*Author:*
**Carrillo Zapata, Daniel**

*Title:*
**Shaping robot swarms through morphogenesis and interactive engagement with society**

# Shaping robot swarms through morphogenesis and interactive engagement with society

By


DANIEL CARRILLO-ZAPATA

Department of Aerospace Engineering
UNIVERSITY OF BRISTOL

Department of Engineering Design and Mathematics
UNIVERSITY OF THE WEST OF ENGLAND


A dissertation submitted to the University of Bristol in accordance with the requirements of the degree of DOCTOR OF PHILOSOPHY in the Faculty of Engineering.

APRIL 2020

Word count: sixty four thousand, eight hundred and sixty-five

Morphogenesis in a robot swarm (*credit to Jerry Wright*).


A fire engine at the Bristol Robotics Lab after a focus group session with local firefighters about robot swarms for fire and rescue services.

# Abstract

Morphogenesis is the process of development of a fully functional biological organism through the spatial self-organisation of millions of cells. The field of morphogenetic engineering takes inspiration from morphogenesis in nature to program self-organised shape formation into functional structures composed of multiple robots to provide them with the high degree of adaptability and robustness seen in biological systems. This would be particularly useful for application areas such as urban search and rescue (to explore a building on fire), utilities inspection (to detect cracks), or nanomedicine (to fight cancer cells), for example.

By taking inspiration from developmental mechanisms discovered in multi-cellular organisms, and building on previous work in the swarm robotics literature, I show the first demonstration of completely self-organised, controllable and functional morphogenesis in large swarms of real, simple robots. A total of 25 experiments with swarms of 300 Kilobots and over 2000 simulations were performed to show the emergent, adaptable and robust shape formation behaviour of robot swarms without the use of any map of the shapes to form, coordinate system or preprogrammed seed robots.

In this thesis, I also show how interactive user and public engagement can have mutual benefits for swarm robotics researchers and society. In particular, two formal studies were carried out. The first study engaged a total of 23 participants from fire brigades through focus groups following the methodology of mutual shaping. In the second study, an educational escape room and group discussion session were developed to engage a total of 52 participants from the general public into the field of swarm robotics and the research done in this thesis. The insight provided by these two studies informs swarm robotics researchers about what needs to be done for robot swarms to be successful and beneficial for society.

# Foreword

This thesis represents almost the last four years of my life. Four years that I have been living and learning in Bristol. Four years that have given me the gift of achieving an immense professional and personal development. I learned to listen to myself, and to follow my passions. I also learned that although a doctorate implies endless hours of dedication to the study, it is not worth sacrificing one's health. Hence, I learned that study time is as important as having time to take care of oneself—through exercise, healthy food and social life. No matter what you do, it will irradiate life and reach further if you also dedicate time to study yourself and find inner peace.

This doctorate made me free. I have been able to choose wisely what I wanted to do in every part of the path—always with my supervisors' guidance. I feel fulfilled because my research has influenced me, some of its potential users and many citizens of this world who have had a glimpse of it. Even if not through me, I am delighted to see that street artists have also honoured Turing patterns in the city that gave me the opportunity to study them. I feel blissed as a result of having connected with a human world through my research in swarm robotics. We shall never forget that technology should be at society's disposal. May we never forget to be humans.

'Clothed with the sun' - EL MAC (Bristol, 2011). Graffiti using a technique of contour lines, reminiscent of Turing patterns.

# Dedication and acknowledgements

Rocío, I would like to start this dedication with you. Thank you for being my companion. Thank you for looking to each other with our inner eyes when taking decisions together. Thank you for embarking in this adventure with the strength and energy of a dancer of life. Thank you for letting us be—free.

Immense thanks to my main supervisor Sabine Hauert for her unconditional support in every single decision during this doctorate, especially for embracing my passion for public engagement. Thank you for sharing your motivation of making useful technology for society with me. I would also like to thank my other supervisors (Alan Winfield, James Sharpe, Luca Giuggioli) for advising me wisely during this time, and the FARSCOPE CDT directors (Arthur Richards, Jonathan Rossiter, Tony Pipe) and EPSRC for having put their trust and funds on me. Also, I would like to thank the FARSCOPE Administration Team and the BRL technicians for their support during this thesis.

I would like to thank my parents, Pedro and Reme, for learning to watch me fly with all the birds above my head. Also, many thanks to all my friends who have listened to me talking passionately about robots, nature, firefighters and theatre (Alberto Sánchez, Alicia Heras, Jose Francisco García, Raúl Alcalá, Teresa Carrillo; Leticia Pardo; Laura Balsalobre, Luis Eduardo Guerrero, María Iglesias, Paco Pousa; Laura Sánchez; Raquel García, Rauad Mohammed; Raquel García; Adrián Solano, Carmen Almela, Carmen Sánchez, Ovidio López). A huge thanks to the Hauert Lab team who have infinitely helped me during my doctorate (Ana Rubio, Edmund Hunt, Emma Milner, Julian Hird, Merihan Alhafnawi, Simon Jones), and my FARSCOPE cohort, especially Chanelle Lee and Krishna Digumarti for their continuous support. Also, I would like to thank Gordon Darling, the BRL porter, who always had an uplifting comment every single morning, afternoon and evening.

Finally, I would like to deeply thank all the people from the Public Engagement Team at the University of Bristol who I have worked with since the last 3 years of my life (Ellie Hart, Hannah Berg, Martha Crean, Suzi Wright, Vivienne Kuh), especially Ellie Cripps and Mireia Bes. Thank you all for opening the doors of public engagement, for awakening my inner reflective and responsible researcher, and for teaching me that researchers can go outside the bubble to engage in a two-way conversation, at the same level, with the people who will be affected by the research. And thank you, Daniel Erice, for making me discover how powerful it is to mix the arts with the science to reach minds through emotions. Our reasoning and emotions is exactly what makes us humans.

## Author's declaration

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Research Degree Programmes and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, the work is the candidate's own work. Work done in collaboration with, or with the assistance of, others, is indicated as such. Any views expressed in the dissertation are those of the author.

SIGNED: .................................................... DATE: ..........................................

# Table of Contents

# List of Tables

**Table** **Page**

# List of Figures

**Figure** **Page**

# List of Abbreviations

| | |
|---|---|
| AP | Anterior-posterior |
| GRN | Gene regulatory network(s) |
| HRI | Human-robot interaction |
| HSI | Human-swarm interaction |
| NSCP | Number of shape characterising points |
| OPI | Object(s) of potential interest |
| PI | Positional information |
| RD | Reaction-diffusion |
| RRI | Responsible research and innovation |
| SI | Shape Index |
| STEM | Science, Technology, Engineering and Maths |

CHAPTER 1

# INTRODUCTION

The study of morphogenesis, which is the study of development of form, dates back to Aristotle (384–322 BCE) in ancient Greece [Davies, 2013]. Aristotle examined the development of the chick embryo over 2000 years ago [Hamburger and Hamilton, 1951]. Since then, scientists have been puzzled about how nature can give form to the immense number of organisms inhabiting our world starting from a single cell, i.e. a fertilised egg, fruit of the union of the male and female sexual cells [Murray, 2003a]. The field of developmental biology has seen many scientists postulating diverse theories to unravel this process. As a consequence, many mechanisms that try to explain the development of multi-cellular organisms from different perspectives have been proposed to date [Salazar-Ciudad et al., 2003]. Indeed, advances in the field of developmental and computational biology in the last few decades have turned morphogenesis into less of a mystery, shaping and deepening our understanding of it [Green and Sharpe, 2015]. This knowledge has motivated the creation of fields such as morphogenetic engineering, which aims to design functional, self-organised structures made of multiple agents with the properties of morphogenesis in biological systems [Doursat et al., 2012]—those of adaptability and robustness. Swarm robotics, where a large number of robots self-organise into complex behaviours only through local interactions and without any leader, is a suitable approach to fulfil the aim of morphogenetic engineering.

This thesis draws inspiration from developmental biology to engineer completely self-organised, controllable and functional morphogenesis in robot swarms. In particular, the first demonstration of a morphogenesis algorithm with swarms of up to 300 real robots developing shapes that are emergent, adaptable and robust is presented. Then, the morphogenesis algorithm is improved by taking inspiration from current hypotheses in the field of developmental biology [Green and Sharpe, 2015] . This improvement led to the development of controllable and functional morphogenesis in robot swarms, resulting in swarms growing shapes to explore their environment and connecting objects of potential interest through the shortest path.

Swarm robotics has been predicted to have an impact on all application areas of robotics during this current decade [Yang et al., 2018]. However, it is also generally impacted by negative public perception driven by science fiction and hyped headlines [Hamann, 2018]. To improve understanding and public trust of swarm robotics, results from one formal study with potential users of robot swarms (fire brigades) and one formal study with the general public, carried out under the umbrella of responsible research and innovation through participatory and interactive approaches, are presented. The aim was to have bidirectional impact on both society (that I define as the combination of users and the general public) and swarm robotics researchers. This is known as *mutual shaping* [Boczkowski, 1999]. As a consequence, the next steps towards deployment of robot swarms in future real-world applications have been identified, and they are presented in this thesis.

## 1.1 Motivation

Swarm robotics is the study of a large number of robots locally interacting with each other to achieve an emergent, global behaviour in a completely self-organised way, i.e. without any central controller [Şahin, 2005]. There are many applications where swarm robotics could be useful. Among them, search and rescue is one of the most promising ones [Murphy et al., 2008; Delmerico et al., 2019]. For such application, the ability of robot swarms to develop spatial shapes to explore an environment [O'Grady et al., 2010], cross gaps [O'Grady et al., 2012] or guide rescue teams [Brambilla et al., 2013] could be advantageous. In nature, examples of these spatially-organising behaviours giving rise to useful functionality can be seen in ants building bridges to cross gaps [Anderson et al., 2002], termites constructing nests and fish schooling for protection from predators [Noirot and Darlington, 2000; Magurran, 1990], organisms such as the slime mould *Physarum polycephalum* modifying their shape to forage, or millions of cells developing limbs and tissues [Raspopovic et al., 2014].

Shape formation in swarm robotics has been successfully demonstrated in large swarms of real robots through self-assembly and self-disassembly [Rubenstein et al., 2014b; Gauci et al., 2018]. However, these approaches were top-down, i.e. they required a map of the desired shape, as well as the construction of a coordinate system and several preprogrammed seed robots. When bottom-up approaches have been used (i.e. no map, coordinate system or preprogrammed seed robots), validation was only done in simulation or using a few dozens real robots. Either top-down or bottom-up, all the work proposed so far has laid the foundations for the development of future robot swarms able to adapt their morphologies to solve real-world problems. As Doursat et al. [2013] postulate, examples of potential applications of morphogenetic systems could be self-assembling mechanical components or robots that adapt to different environments, self-morphing particle swarms that penetrate through tissue to deliver drugs, self-coding software that optimises performance, and even self-constructing buildings or self-managing energy grids.

If we want to see swarms of robots in the physical realm, we need new algorithmic contributions that embed the reliability in solving morphogenetic tasks of self-organised biological systems into the architecture of our human-made systems. This thesis proposes new morphogenesis algorithms that combine mechanisms suggested by the developmental biology community with some of the state of the art approaches proposed by the swarm robotics community. By bringing these two worlds together, morphogenetic robot swarms can be a step closer to becoming a real solution for society.

Because the technology will be ready to move beyond laboratories to the real world in the not-so-distant future, engagement of society in the research and development of swarm robotics is needed now more than ever to maximise its success. It is important to acknowledge that swarm robotics suffers from negative public perception as a result of science fiction [Hamann, 2018], even if this technology has been foreseen to have a substantial impact during this decade [Yang et al., 2018]. Therefore, it is also important to engage with potential users to adequate morphogenetic swarm robotics technology to their needs [Delmerico et al., 2019]. Furthermore, if we want to successfully deploy robot swarms in the real world, it is crucial to engage the general public in a dialogue about the use of swarm robotics technology and their attitudes/concerns about this. Only by increasing user and public acceptance and trust can we realise the economic and societal benefits of technology [Winfield and Jirotka, 2018]. However, there exists a gap in the literature regarding engagement of users, stakeholders and the general public in the topic of swarm robotics, hence providing an opportunity for the studies presented in this thesis to open up a bidirectional dialogue with society.

## 1.2 Research questions

The primary aim of this thesis is to achieve completely self-organised, bottom-up morphogenesis leading to emergent, adaptable, robust, controllable, and functional shapes in large robot swarms of real robots. The secondary aim is to engage potential users of the technology developed in this thesis and the general public to understand their attitudes, hopes, concerns and requirements of robot swarms. Taking these into account, the main research questions that this thesis addresses are the following:

- Can large swarms of real robots form fully self-organised shapes that are adaptable and robust?
- Can these shapes be controlled and functional?
- What are the main needs, attitudes and concerns from fire brigades with respect to robot swarms being used to assist them in fire & rescue missions?
- What are the main attitudes and concerns from the general public about robot swarms being applied to solve real-world problems?

## 1.3 List of contributions

The process of addressing the previous research questions has led to the following contributions in this thesis:

- First demonstration of a completely self-organised morphogenesis algorithm by taking inspiration from mechanisms discovered in multi-cellular development. The swarm behaviours were shown to be emergent, adaptable and robust in simulation and real swarms of 300 Kilobots (chapter 3). Results were published in the Science Robotics journal. This contribution was the result of a collaboration with the European Molecular Biology Laboratory led by James Sharpe in Barcelona, Spain.

- Development of a controllable morphogenesis algorithm by taking inspiration from current hypotheses in developmental biology. The algorithm was shown to be emergent, scalable, robust, and able to get around obstacles in simulation and real swarms of 300 Kilobots (chapter 4). Results were published in IEEE Robotics and Automation Letters, and presented at IEEE IROS 2019 conference.

- Development of a functional morphogenesis algorithm able to explore an environment and connect objects of potential interest through the shortest path to guide users, as an example of swarm-guided navigation. The algorithm was shown with swarms of 250 robots in simulation (chapter 4).

- Formal study following the framework of mutual shaping with 23 participants from fire brigades to understand their profession, challenges, needs and attitudes towards robots in general and robot swarms in particular for fire and rescue, as well as their level of acceptance and opinions on autonomy and their involvement during the research and development process (chapter 5). This study is the first of its kind that formally engages users by understanding their attitudes, hopes, concerns and requirements about the topic of swarm robotics in particular. I show that fire brigades are generally positive about the use of semi-autonomous robot swarms, and that they mainly need them to be reliable, safe, efficient and transparent for firefighters to trust them. Results were published in the Special Issue on Designing Self-Organization in the Physical Realm of Frontiers in Robotics and AI - Multi Robot Systems journal.

- Formal study following the framework of gamification with 52 participants from the general public to teach about the principles of swarm robotics and the research on morphogenesis carried out in this thesis, as well as to understand the general public's attitudes towards being assisted by robot swarms at work/home, their benefits versus risks for society, potential applications, opportunities, concerns and actions that they think researchers, companies and/or governments should be doing to push robot swarms into the real world (chapter 5).

This study is the first of its kind that formally engages the general public by understanding their attitudes, hopes, concerns and requirements about the topic of swarm robotics in particular. A swarm-robotics-themed educational escape room and a group discussion session were developed for this study. I show that participants learned about swarm robotics through playing the escape room, and that it made unsure players have an opinion about swarm robotics after the game. I also show the wide range of beneficial applications that the general public came up with, as well as their main concerns about robot swarms (e.g. misuse, sustainability, transparency, safety, lack of regulation) expressed in the group discussion session.

- Co-development of a workshop on bio-inspiration for primary and secondary school students, which was facilitated to 219 students in France. We show a direct correlation between having fun during the workshop and both learning something new and becoming more interested in STEM subjects (Science, Technology, Engineering and Maths), as well as a direct correlation between the language being easily understood and the ability to have fun, and therefore learning and engaging in STEM. Results were presented at the International Conference on Robotics and Education RiE 2019.

## 1.4 List of publications

The following peer-reviewed publications have been made during this thesis:

- **(Co-)First author**:
  - (**Journal**) Slavkov, I., Carrillo-Zapata, D., Carranza, N., Diego, X., Jansson, F., Kaandorp, J., Hauert, S., & Sharpe, J. (2018). Morphogenesis in robot swarms. Science Robotics, 3(25), eaau9178.
  - (**Journal**) Carrillo-Zapata, D., Sharpe, J., Winfield, A. F. T., Giuggioli, L., & Hauert, S. (2019). Toward controllable morphogenesis in large robot swarms. IEEE Robotics and Automation Letters, 4(4), pp.. 3386–3393. **This work was also presented at the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2019).**
  - (**Conference**) Carrillo-Zapata, D., Lee, C., Digumarti, K. M., Hauert, S., & Boushel, C. (2019). Lessons from Delivering a STEM Workshop Using Educational Robots Given Language Limitations. In International Conference on Robotics and Education RiE 2019 (pp. 284–295). Springer, Cham.
  - (**Journal**) Carrillo-Zapata, D., Milner, E., Hird, J., Tzoumas, G., Vardanega, P., Sooriyabandara, M., Giuliani, M., Winfield, A. F. T., & Hauert, S. (2020). Mutual Shaping in Swarm Robotics: User Studies in Fire and Rescue, Storage Organization, and Bridge Inspection. Frontiers in Robotics and AI, 7(53).

## 1.5 List of outreach and public engagement activities

Below is a list of the outreach and public engagement activities carried out during this thesis:

- **15 March, 2016**: Facilitation of a robotic session at Bradley Stoke Library for kids aged 5+. The activity consisted of demonstrating the robot Thymio, plus teaching how to program it.
- **29 June, 2016**: The second cohort of students of the FARSCOPE CDT took the robots that we designed as part of our group project (the *tortoises*) to @Bristol, the science museum in Bristol, for the audience to interact with them.
- **13 July, 2016**: The second cohort of students of the FARSCOPE CDT devised several activities aimed to involve students from Abbeywood school in changing the software and hardware of the *tortoise* robots.
- **13-17 November, 2017**: Chosen by the British Council as part of their *Science in Schools* programme to deliver nine robotic workshops to a total of 219 pupils aged 12-14 in the French regions of Nancy and Metz during a week. Two other FARSCOPE students came as well. We organised a workshop around bio-mimicry in robotics with interactive parts where pupils could program some animals' behaviours on Thymio robots.
- **25 October, 2017 - June 15, 2018**: One of the participants in the second cohort of the European Project PERFORM (grant agreement No 665826) with the University of Bristol Centre for Public Engagement. During the first part of the programme, we, post-graduate researchers, received training on ethics and responsible research and innovation (RRI), as well as communication skills through performing arts. During the second part, I had four sessions at Bristol Free School with two students aged 14 who I helped develop a science busk about my research topic. Finally, I was chosen to participate in the project conference held at UNESCO in Paris on June 14-15, 2018. There I took part in three panel sessions and one science busking session. Over 200 people attended the conference, from secondary school students and teachers from Paris and Barcelona, to researchers and academics mixing art with science communication and promoting RRI.
- **31 January, 2018**: Engagement Bites talk about my experience with the British Council programme for University of Bristol audience.
- **12 February, 2018**: I helped three classes from St. Werburghs primary school to enter an engineering competition where the students had to come up with an invention inspired by my talk on swarm robotics.
- **19 May, 2018**: All PhD students from the second cohort of the FARSCOPE CDT held an event as part of Creative Reactions weeks where we played improvisation theatre games to

start conversations about robots. The aim was to know what the attendees (mostly young adults) think about robots in a very interactive way.

- **25 July, 2018**: Speaker at *Telling Stories about Robots* event at We the Curious (the rebranded science museum of Bristol), where we discussed the topics of militarisation, attachment and evolution of robots through films and literature. I also had the opportunity to perform the busk I developed during the PERFORM project.

- **26 July, 2018**: One of the experts in the *After Hours: What if I could change the world?* adults-only session at We the Curious, where the FARSCOPE CDT demoed some Thymios and Kilobots, and gathered responses to the public survey that the FARSCOPE CDT carried out to understand people's concerns about robots.

- **28 September, 2018**: Storytelling of my research during Futures at We the Curious on the European Researchers' Night 2018.

- **28 October, 2018**: Invited by the Royal Society for a demo of swarm robotics for their *You and AI* series as part of Manchester Science Festival.

- **13 November, 2018**: Workshop about my research in swarm robotics and responsible research and innovation at the Faculty of Computer Science, University of Murcia, Spain (faculty where I did my undergraduate degree).

- **13 March, 2019**: Engagement Bites talk about my experience in the PERFORM project.

- **14 March, 2019**: The *Black Researchers in STEM* group were awarded funding to run a robotics inclusion day during British Science Week. The inclusion day saw around 30 schoolchildren from multicultural backgrounds visit the BRL for a robotics workshop that I facilitated.

- **4, 15 and 22 April, 2019**: Organised three focus groups with fire brigades, as explained in chapter 5.

- **8 and 15 May, 2019**: Participation in the project *Shape Your World* as part of the *Future Quest* umbrella project with the Centre for Public Engagement from the University of Bristol.

- **27-28 September, 2019**: Prototyping of *Swarm Escape!* escape room puzzles at SS Great Britain and We the Curious during the European Researchers' Night 2019.

- **16-18 October, 2019**: Facilitation of *Swarm Escape!* escape room at Millennium Square and *Robot Swarms in our Cities* session at Watershed as part of the Festival of Ideas 2019. More details given in chapter 5.

## 1.6 Structure of the thesis

The rest of this thesis has been structured as follows:

- Chapter 2 provides a background on morphogenesis and pattern formation from a developmental biology perspective to explain past and current theories of multi-cellular development. Then, a review of the state of the art in morphogenesis in swarm robotics and morphogenetic engineering is given. Finally, the state of the art in society engagement in swarm robotics is reviewed.

- Chapter 3 presents a novel morphogenesis algorithm based on patterning and migration to demonstrate completely self-organised shape formation, as well as 20 experiments carried out with swarms of 300 real robots and 121 simulations to show emergence of shapes, adaptability to different initial configurations and robustness to minor and major damage.

- Chapter 4 presents a novel morphogenesis algorithm based on patterning, migration and local gradients to demonstrate completely self-organised, controllable shape formation, as well as 5 experiments on emergence, scalability, robustness and response to obstacles with swarms of 300 robots and over 2000 simulations. Finally, this chapter concludes with an extension to the controllable morphogenesis algorithm whereby functionality in the form of swarm-guided navigation is shown in a simulated swarm of 250 robots.

- Chapter 5 presents the main findings of the mutual shaping study with 23 potential users of robot swarms (firefighters) with respect to their attitudes towards the use of robot swarms to assist them. The main findings of another study with the 52 participants of general public through an educational escape room and a group discussion session are also given in this chapter. Finally, common findings from both studies are described in an effort to push successful deployment of robot swarms into the real world.

- Chapter 6 gives a summary of all the contributions of this thesis, and lists their limitations and future work related to morphogenetic robot swarms and engagement of society. In addition, a brief reflection on environmental, economic, societal and cultural impact is given at the end of this chapter.

- Appendix describes improvements on the Kilobots firmware to filter out background noise and send the robots to low-power consumption mode when they detect their batteries are running out.

# Chapter 2

## Background and state of the art

From thousands of birds flocking together or ants creating trails, to embryo development or termites building complicated-but-organised structures, the spatially-organising behaviours seen in biological systems are a promising source of inspiration in swarm robotics. In particular, the shape formation process seen in the development of tissues and organs in multicellular organisms (known as *morphogenesis*), shows one of the most desired features of robot swarms: programmable self-organisation. During morphogenesis, millions of cells self-organise into structured, spatial patterns to create a fully-functional organism in a completely decentralised fashion, where guidance is genetically encoded inside cells—the genetic program. This morphogenetic process exhibits a high degree of adaptability, self-regulation and robustness. These properties could be particularly beneficial for large robot swarms of simple robots to overcome their stochastic nature.

The recent field of morphogenetic engineering was founded to design multi-agent, autonomous systems able to self-organise into controllable and functional shapes without any central controller. The first part of this thesis precisely focuses on designing completely self-organised, controllable and functional shape formation algorithms for large swarms of simple, real robots. For that, multi-cellular morphogenesis is taken as inspiration. Bio-inspiration is used as a reference for the design of the algorithms. Therefore, the goal is not to imitate the morphogenesis process of biological systems, but to engineer self-organising robot swarms that show emergent, adaptable and robust shape-formation capabilities, as also seen in nature. Such morphogenetic robot swarms could be used in architecture (e.g. self-constructing buildings that physically adapt to spatial conditions), urban search and rescue (e.g. to overcome obstacles while exploring a disaster environment), or biomedicine (e.g. drugs diffusing through tissue), for example.

Due to the wide array of future applications of morphogenetic robot swarms, it is crucial to engage relevant users and stakeholders, as well as the general public, in the development of this swarm robotics technology. The second part of this thesis focuses on understanding their needs and concerns, and incorporating society in the process. By doing so, morphogenetic robot swarms are bound to be successfully deployed in the future.

In this chapter, a background on morphogenesis from a developmental biology perspective is given to explain the current understanding and theories of multi-cellular development as well as a review of metrics that have been used in the literature to characterise morphologies, followed by sections giving an overview of the state of the art on morphogenesis-inspired, human-engineered robotic systems under the umbrella of morphogenetic engineering and swarm robotics. Finally, an overview of the state of the art on user and public engagement work exploring how to take robot swarms into society is given.

## 2.1 Morphogenesis in nature from a developmental biology perspective

The term *morphogenesis* comes from the Greek roots *morphê* (form) and *genesis* (creation). In the field of developmental biology, morphogenesis is the *"development of pattern and form"* [Murray, 2003b], i.e. it is the process by which cells spatially and temporally organise to eventually grow a three-dimensional, structured organism—this being an animal, a plant, fungi, etc. Morphogenesis occurs during *embryogenesis*, which concerns the whole developmental process of an embryo from a simple, unicellular fertilised egg (the zygote), to the creation of a complex multi-cellular organism with *"hundreds of distinct differentiated cell types"* [Sagner and Briscoe, 2017]. Morphogenesis is a crucial aspect in embryo development because cells position themselves in the right place at the right time in an orchestrated way [Briscoe, 2019] to define genetic territories, which are regions containing cells with equivalent states of gene expression (the genetic programs) [Salazar-Ciudad et al., 2003]. Thus, cells differentiate, i.e. they run their genetic program by making fate choices, depending on their spatial locations [Murray, 2003b]. The fundamental question that developmental biology addresses is how these genetic programs are triggered in the right spatiotemporal order to create patterned tissues [Sagner and Briscoe, 2017]. In other words, developmental biology tries to understand what mechanisms are in place during embryo development and how they self-organise to produce a functional organism [Jaeger and Sharpe, 2014].

The organisation into regions of cells alike is essentially a process of pattern formation, and this will be the main focus of this section. One of the first examples of pattern formation during morphogenesis of a biological organism was discovered in the *Drosophila* fruit fly. Wolfgang Driever and Christiane Nüsslein-Volhard discovered in the late 80s that the maternal *Bicoid* gene[1] releases a protein named *bcd* that defines a gradient along the anterior-posterior (AP)

[1] Maternal genes are those present in the fertilised egg or embryo and drive development following the maternal genetic code until the zygote uses its own genetic code, i.e. both the maternal and parental genetic codes [Marlow, 2010].

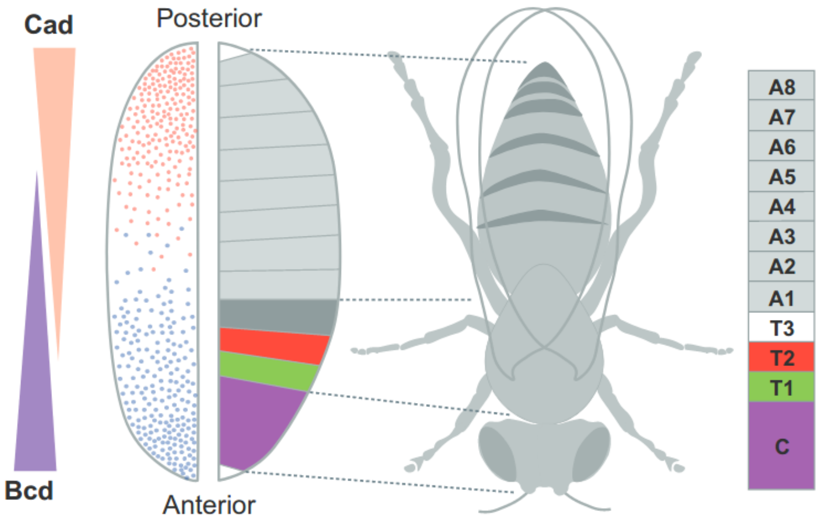


Figure 2.1: Patterning in the *Drosophila* fruit fly. Proteins bcd and cad define gradients that target different genes depending on the concentration of proteins. Hence, a pattern of different genetic territories emerges. These territories will eventually become the cephalic (C), thoracic (T1-T3) and abdominal (A1-A8) segments. *Image reproduced from Briscoe and Small [2015]. Article is licensed under CC BY 3.0.*

axis of the early embryo of *Drosophila* [Driever and Nüsslein-Volhard, 1988a,b]. Other authors later discovered that the *caudal* protein (cad) creates an anti-gradient in parallel to bcd [Chan and Struhl, 1997; Niessing et al., 2002]. Together, these two gradients *"target genes that create boundaries at defined positions along the AP axis"* [Briscoe and Small, 2015]. As a result, a pattern composed of different genetic territories defined by the concentration of proteins along the AP axis emerge to divide the body plan into regions that will eventually become the cephalic (C), thoracic (T1-T3) and abdominal (A1-A8) segments after their particular gene expression, as illustrated in figure 2.1.

Patterning of tissues and organs mostly takes place during the early stages of embryonic development [Kimmel et al., 1995]. Indeed, a human embryo has developed 90% of all their structures by the $8^{th}$ week of development ($10^{th}$ week of gestation), even if the embryo is the size of half the length of an adult thumb at that age [O'Rahilly, 1979]. During this early period, cells in the human zygote and embryo undergo a series of spatial changes that can be categorised in four main stages: cleavage, blastulation, gastrulation and organogenesis [Gilbert, 2000]. These stages are in fact common across vertebrates and mammals, and some of them can also be found in plant embryogenesis [Baroux and Grossniklaus, 2015]. During cleavage, the zygote—the fusion of the maternal and paternal reproductive cells known as gametes—divides exponentially through mitosis to form a mass of *blastomere* cells known as *morula*. The size of the morula is the same as the zygote because of the protective layer of proteins that limits growth, known as *zona pellucida*. A simple patterning can be observed in the morula during blastulation. Outer cells form

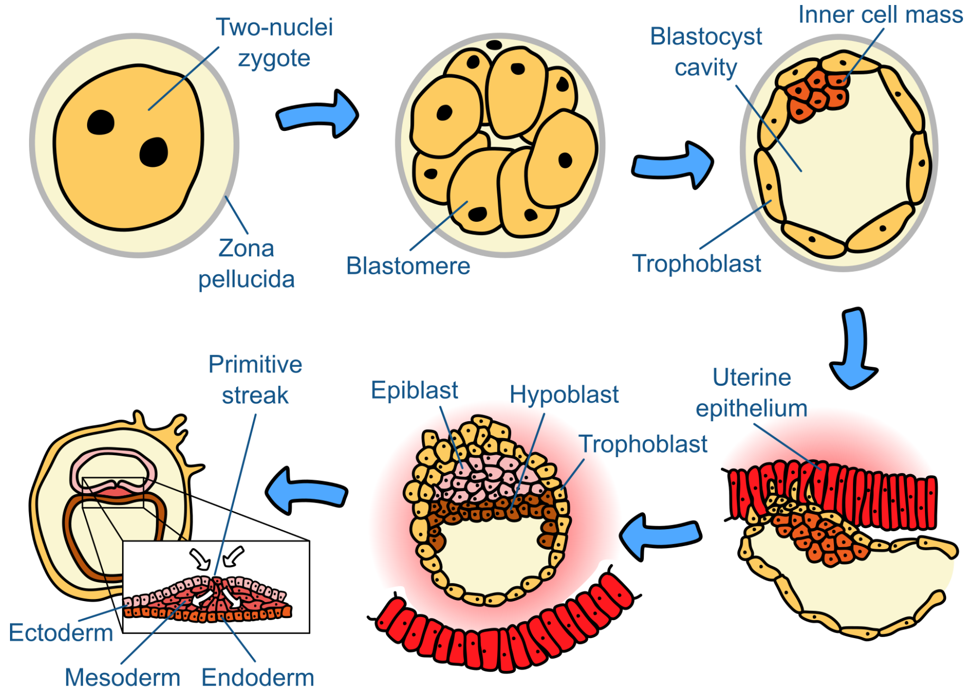


Figure 2.2: Early stages of human embryogenesis. The fertilised zygote starts division into blastomeres to form the morula. Cells in the morula spatially organise into the blastocyst, which is composed of an outer layer (trophoblast), an inner cell mass on one side and a cavity. Then, cells in the trophoblast adhere to the wall of the mother's uterus through the uterine epithelium cells. The blastocyst continues pattern formation by the inner cell mass differentiating into two layers: the epiblast and hypoblast. A cut in the epiblast then appears (the primitive streak), provoking a flow of cells that creates three more layers: the ectoderm, mesoderm and endoderm. *Image modified from Zephyris's Human Embryogenesis image, which is licensed under CC BY 3.0.*

a spherical layer known as *trophoblast*. Some of its cells will eventually develop structures for the implantation in the maternal uterus to receive nutrients (e.g. the umbilical cord). The inner cell mass in the morula is pushed to one side of it, forming a hollow ball known as *blastocyst*. Another process of patterning then takes place. The inner cell mass differentiates further into two disc layers, known as the *epiblast* and the *hypoblast*. During gastrulation, the epiblast is cut by the *primitive streak*, which represents the midline of the body that separates the left and right sides. As soon as this happens, cells start to flow from the epiblast to the space between the epiblast and the hypoblast. As a consequence, three new layers are formed: the *ectodermn*, the *mesoderm* and the *endoderm*. These three layers are known as *germ layers*. During organogenesis, the ectoderm will form the external parts of the body (e.g. epidermis, hair, neurons, spinal cord), the mesoderm will form the middle parts (e.g. muscle, circulatory system, bones), and the endoderm will form the internal parts (e.g. rest of organs). These first stages are illustrated in figure 2.2. Later stages of development (e.g. fetal development in viviparous animals) then focus on massive growth of such structures [Jirásek, 2012].

As stated by Salazar-Ciudad et al. [2003] and reflected above, *"the development of an organism can be described as transformation from one set of patterns to another set of patterns"*, i.e. from one spatial arrangement of cells to another over time. The understanding of these patterning and morphogenesis processes mostly came from theoretical hypotheses until the late twentieth century [Horder, 2010]. Until then, biologists could only rely on *"pencil-and-paper analysis"* [Morelli et al., 2012]. However, advances in molecular genetics and computational modelling made the field of developmental biology live its golden age during the period 1980–2000 [St Johnston, 2015]. Computational models allowed biologists to test and refine developmental theories by predicting the observed experimental data [Morelli et al., 2012]. Computer modelling is still key to advance in the understanding of biological processes during embryogenesis. Indeed, the current resurrection of developmental biology is driven by the development of more sophisticated computational models based on machine learning that process the enormous amount of data collected these days [Angermueller et al., 2016].

### 2.1.1 Mechanisms of pattern formation

Experimental evidence has led to the discovery of several mechanisms that produce patterning, as reviewed by Salazar-Ciudad et al. [2003]. The authors define such mechanisms as gene regulatory networks (GRN) that regulate specific cell behaviours (e.g. cell division, cell death, cell-to-cell signalling, etc.), hence producing the transformation from one spatial pattern to another [Salazar-Ciudad, 2010]. A GRN describes the network that connects some genes in the cell (the *regulatory genes*) with other regulatory genes, through the proteins and molecular products they synthesize, to regulate their gene expression [Davidson, 2010; Peter and Davidson, 2015]. In a nutshell, GRN are the control logic of the cell that decide its fate, i.e. its behaviour/function. The genomic control produced by GRN is of high importance for biological processes [Karlebach and Shamir, 2008], because they determine the spatiotemporal patterning of cells to develop the formation of the organism during embryogenesis [Peter and Davidson, 2015]. Salazar-Ciudad et al. [2003] classify the GRN developmental mechanisms of pattern formation in three types: cell autonomous mechanisms, inductive mechanisms and morphogenetic mechanisms. It is important to highlight that this classification is phenomenological in the sense that it is based on *"correlations representing similarities between observables, instead of causal explanation of how the observed regularities arise"*, as Jaeger and Sharpe [2014] state. However, this classification of mechanisms is indeed a great source of inspiration for shape formation in robot swarms, as some of the mechanisms can be easily implemented in robots. Below is a description of each one, accompanied by some of their biological examples discovered so far.

Through mitosis, cells can perform pattern formation autonomously. For example, in organisms such as mice, frogs and flies, the different spatial regions with different genetic code in the fertilised egg are explained by the generation and further division of polarised and non-polarised cells during cleavage [Fleming and Johnson, 1988; Müller, 2001]. Polarised cells are those with a spatial asymmetry in their internal material. In the case of organisms such as nematodes (a type of worm), polarised cells divide into two distinct cells because the genetic material is divided asymmetrically [Bowerman and Shelton, 1999].

Pattern formation following inductive mechanisms are those arising from cell-to-cell communication through diffusible molecules, i.e. chemical signals secreted by cells that act as messages. Morphogenesis following this type of patterning is also known as *chemical morphogenesis*. As a result of the interpretation of the messages, cells can change their internal genetic state, hence creating a pattern (as in the example of *Drosophila* in figure 2.1). Researchers have discovered that many of these signals act as morphogens [Briscoe and Small, 2015; Gurdon and Bourillot, 2001]. The term *morphogen* was first coined by Turing [1952]. It refers to a chemical that can be diffused over long distances—hence creating a gradient—, and with the ability to trigger cell fates based on its particular concentration when it reaches the cell. Morphogen gradients have been found to play a key role in providing asymmetry in embryos to define axis in the body plan [Briscoe and Small, 2015], e.g. anterior-posterior axis, proximal-distal axis, etc. In experiments with mutant mouse embryos with the *Sonic hedgehog* morphogen protein (shh) suppressed, some cells from the dorsal neural tube (at the back) travel to the ventral neural tube (at the front) [Chiang et al., 1996; Litingtung and Chiang, 2000; Wijgerde et al., 2002], as seen in [Briscoe and Small, 2015]. The symmetry-breaking phenomenon of axis formation is indeed crucial for a proper development of the embryo [Levin, 2005]. The inductive mechanism whereby a genetic territory or a single cell sends a morphogen to others but does not respond to the morphogens produced by the cells it communicates with, is called *hierarchic* (e.g. maternal morphogens). On the other hand, when cells are affected by the morphogen response of other cells, this is known as *emergent*. The emergent inductive mechanism of reaction-diffusion, which describes the spontaneous self-organisation of morphogens into spatial patterns, has been experimentally observed during several processes of embryogenesis. This mechanism will be further described in §2.1.2.2.

Morphogenetic mechanisms are those that modify the spatial pattern by affecting its form without changing the genetic state of the cells, i.e. pattern formation is achieved by mechanical means. This is the reason that morphogenesis following this type of patterning is also known as *mechanical morphogenesis*. One example of morphogenetic mechanism is *migration*, which implies movement of cells from one location to another. Migration has been found to play a key role in pattern formation during gastrulation [Keller, 2005]. For example, cells in the neural crest of mice, arising from the ectoderm, have been found to migrate laterally and in the direction of the abdomen following a chemical gradient of *fibroblast growth factors* (FGF) proteins [Kubota and Ito, 2000]. This process is know as *chemotaxis*. Another important morphogenetic mechanism is *apoptosis*—programmed cell death. The development of several structures in organisms depends on the spatiotemporal death of some cells. In turn, this process also depends on the cells surrounding the cells performing apoptosis, hence not being a cell autonomous mechanism. In vertebrates, apoptosis is key for the development of the nervous system [Oppenheim, 1991; Yuan and Yankner, 2000]. Furthermore, the patterning of digits in the limbs of mice, ducks and bats has been discovered to be driven by sculpting the excess of interdigit tissue through apoptosis [Chen and Zhao, 1998; Suzanne and Steller, 2013]. Another interesting morphogenetic mechanism is *cell adhesion*, by which cells adhere to each other to form tissue layers and compartments with the corresponding boundaries between them [Amack and Manning, 2012]. Recent experimental evidence in *Drosophila* has shown that such layers are formed by inhibition of adhesion at the boundaries by signalling molecules, i.e. cells adhere to each other until they are signalled to stop adhering, thus creating layered tissues [Monier et al., 2011]. Other morphogenetic mechanisms described by Salazar-Ciudad et al. [2003], but with less direct implementation in robotics, are *directed mitosis*, whereby cells divide in a specific direction, *differential growth*, whereby cells with different genetic states divide at different rates, or *matrix modification*, whereby changes in the molecular structure of the extracellular matrix produce a spatial pattern (e.g. buds).

### 2.1.2 Combined mechanisms of pattern formation

The complex process of development of a fully functional multi-cellular organism often combines inductive and morphogenetic mechanisms during embryogenesis [Salazar-Ciudad et al., 2003]. There are two theories of pattern formation during embryogenesis that have stood out to the present: positional information and reaction-diffusion. I refer to these theories as combined mechanisms because they postulate that cells make fate choices (i.e. morphogenetic mechanisms) as a response to morphogens produced by other cells (i.e. inductive mechanisms). In the case of positional information, the focus is put on the way cells interpret such morphogen signals to form patterns, whereas reaction-diffusions focuses on the self-organisation of the signals into patterns. Each theory is described in more detail below.

#### 2.1.2.1 Positional information

Positional information, also known as the French Flag Model, was devised by Lewis Wolpert in the late 60s and early 70s [Wolpert, 1969, 1971]. Wolpert theorised that cells make fate choices depending on the information about their location (i.e. position). In fact, positional information acts similarly to a coordinate system that is available to cells in the form of a chemical gradient of morphogens that cells react to [Murray, 2003b]. The key idea in Wolpert's theory is that the morphogen gradient giving positional information (known as *pre-pattern*) and the final pattern are produced by two different processes. For that, he introduced the concept of the *interpretation step*, whereby cells interpret the morphogen gradient and make the appropriate fate choice (e.g. differentiation, migration or programmed cell death) depending on the concentration of such morphogens, thus, giving rise to any arbitrary pattern, in principle [Green and Sharpe, 2015]. The advantage of this simple theory is that different organisms can preserve the same positional information mechanism in the form of morphogen gradients, while having different

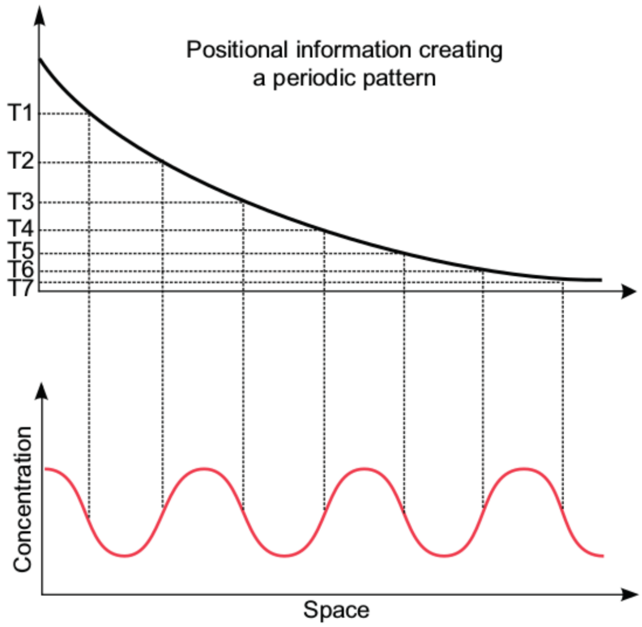


Figure 2.3: A possible emergence of a spatially periodic pattern through positional information in one dimension. A smooth, monotonic chemical gradient defines several thresholds (T1-T7) that cells can interpret to spatially organise into a periodic pattern. *Image reproduced from Green and Sharpe [2015]. Article is licensed under CC BY 3.0.*

interpretations to it (for example, among species or to grow different tissues in the same organism) [Green and Sharpe, 2015]. This is in fact suitable for evolutionary processes, as the positional information mechanism can be reused—interpretation is what would evolve.

In the example of a possible emergence of a spatially periodic pattern through positional information shown in figure 2.3, a morphogen gradient defines several thresholds of concentration (first step: pre-pattern) that cells can interpret accordingly (second step: interpretation). It is important to highlight that Wolpert did not specify how such initial morphogen gradient is created, but how cells use it. Successful demonstration of positional information in nature came around twenty years after it was postulated. The most remarkable example was the discovery that, firstly, there exists a protein gradient created by the *Bicoid* gene [Driever and Nüsslein-Volhard, 1988b], and secondly, that it defines the anterior and posterior axis during the morphogenesis of *Drosophila* (a common fruit fly) [Driever and Nüsslein-Volhard, 1988a]. Another crucial discovery was made by Akam [1989]. He found that *"Hox genes encode positional information along the anterior-posterior axis of all animals"* (as read in [Green and Sharpe, 2015]). This result confirmed that evolution shapes interpretation of the morphogen gradients to create different morphologies in response to the same enconding of positional information. Another successful example of positional information was the discovery that several proteins can trigger different cell fates depending on their concentration in *Xenopus* (a type of frog) [Green and Smith, 1991]. Despite these successful examples in favour of positional information, recent studies suggest that this

Figure 2.4: Examples of patterns on animal skin. From left to right, top to bottom: cheetah (*credit to Wayne S. Grazio*), tiger (*credit to Leszek Leszczynski*), zebra (*credit to Lindsey Elliott*), giraffe (*credit to Daniel Ramirez*), butterfly (*credit to JG D70s*), California Kingsnake (*credit to J. Maughn*), giant Puffer fish (*credit to Chiswick Chap*), angelfish (*credit to Albert kok*), and Conus Omaria shell (*credit to Anders Sandberg*). *All images are licensed under Creative Commons.*

theory is not responsible alone for morphogenesis. Indeed, reaction-diffusion can also play its part in the process.

#### 2.1.2.2 Reaction-diffusion

Several years before Lewis Wolpert developed his theory of morphogenesis, Alan Turing, in his only contribution to biology, envisaged that pattern formation may be spontaneously caused by self-organising cells in response to morphogen gradients interacting with each other and diffusing through space [Turing, 1952]. Alan Turing, a great mathematician and the father of computer science and artificial intelligence, was puzzled by the enormous array of patterns seen in biological systems (e.g. animal skin, as show in figure 2.4). In such examples, it is obvious that patterns possess some structure—they are not just random patterns. In his theory, he addressed a different fundamental problem than that of Wolpert's. Turing was interested in finding a mathematical model that could generate spontaneous, self-organised spatial patterns from a quasi-homogeneous field of molecules (with some small and random fluctuations). He mathematically (and elegantly) described a system of equations called reaction-diffusion that could precisely do that. Murray [2003a] wrote its general form as follows:

$$\frac{\delta c}{\delta t} = f(c) + D\nabla^2 c \tag{2.1}$$

In equation 2.1, $\delta c$ represents the change in concentration of the vector of morphogen concentrations $c$ over time ($\delta t$), $f$ represents the reaction kinetics of the molecules (i.e. how they interact with each other by creating/destroying themselves), and $D$ is the diagonal matrix of diffusion coefficients of such molecules, which diffuse through space following dynamics defined by $\nabla^2$.

Turing mainly focused on activator-inhibitor reaction-diffusion systems of two morphogens, one being an activator (able to enhance secretion of itself and/or other morphogens by cells), and the other one being an inhibitor (able to inhibit production of itself and/or other molecules by cells). The key idea in Turing's theory is that patterning is indeed induced by diffusion of such morphogens—what he called *diffusion-driven instability*. This is a very counterintuitive concept, as diffusion usually increases the entropy of the system, effectively moving away from order (e.g. a drop of dye in water). As Murray [2003b] paraphrases from Turing's work, in the absence of diffusion, the system *"tends to a linearly stable uniform steady state"*. When diffusion is added to the system, a spatial pattern can arise through a symmetry-breaking chain of reactions starting from small molecular fluctuations. For a pattern to emerge, it is crucial that the rate of diffusion of the inhibitor is much greater than that of the activator to achieve short-range activation and long-range lateral inhibition. These two properties are indeed the fundamentals of the reaction-diffusion model. The crucial role that short-range activation and long-range lateral inhibition play in the patterning process, mainly induced by diffusion, is shown in the example of the possible emergence of a spatially periodic pattern through reaction-diffusion of figure 2.5. More specifically, short-range activation and long-range inhibition are the result of having a higher diffusion rate of the inhibitor, which causes the peaks of the activating molecule to stabilise as well as preventing other peaks from forming immediately next to the others [Green and Sharpe, 2015].

Turing was able to demonstrate by manual simulations that reaction-diffusion systems of two or three morphogens could produce different patterns, from stable periodic patterns such as spots, stripes and labyrinths, to travelling waves and oscillations. Unfortunately, he died before seeing biological proofs of his theory. Three decades had to pass until replications of biological patterns were achieved using the reaction-diffusion model [Meinhardt, 1982]. This success was the precursor of even more proof in favour of reaction-diffusion as far as pattern replication was concerned [Murray, 2003b; Meinhardt, 2009], as seen in [Kondo and Miura, 2010].

Reaction-diffusion is fundamentally different from the positional information idea that the pattern emerges from the interpretation of morphogen gradients made by cells. Reaction-diffusion is a completely self-organised process, and it has the property of self-regulation, a property that positional information lacks [Kondo and Miura, 2010]. However, Turing's theory was shadowed by Wolpert's during many years, as well as seen as the antithesis of positional information. Apart from being hard to believe due to the counterintuitive notion of diffusion-driven patterning, no

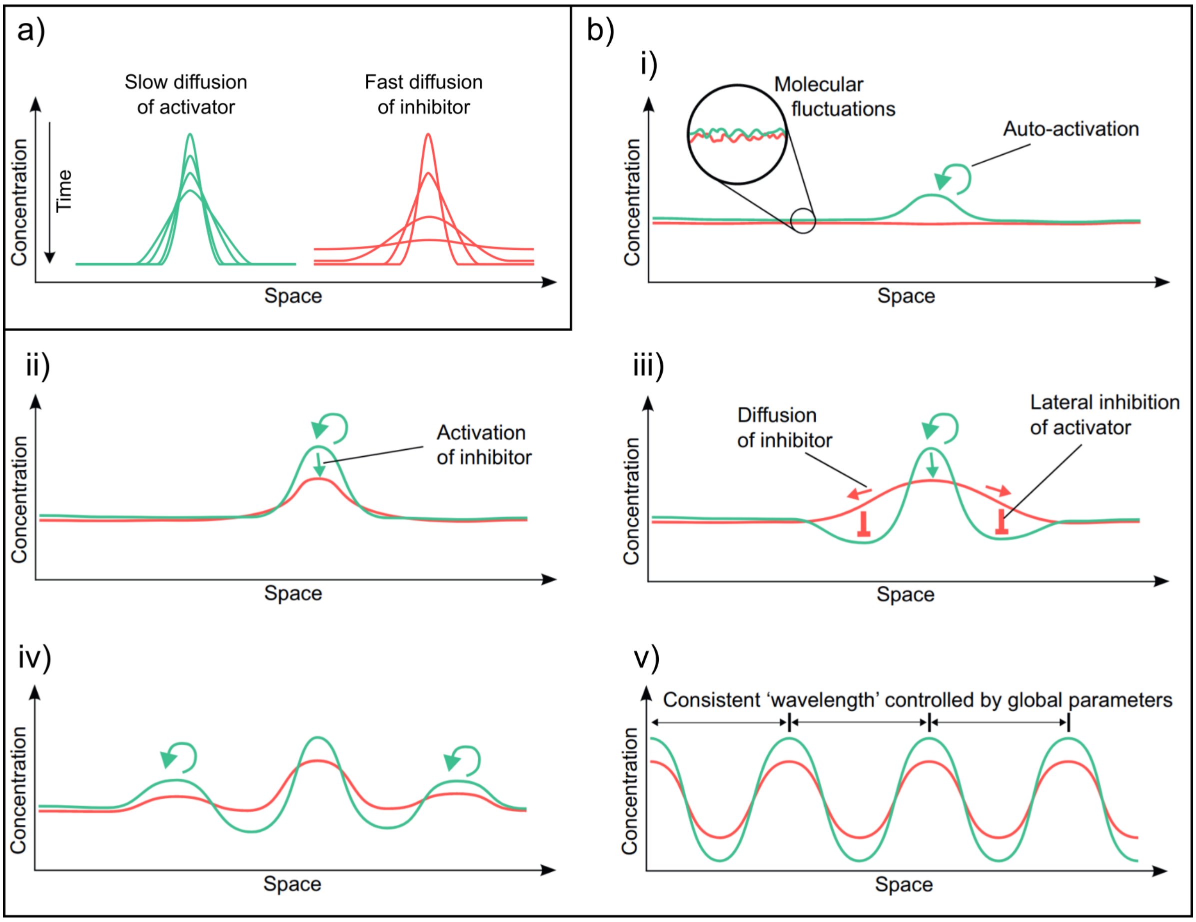


Figure 2.5: A possible emergence of a spatially periodic pattern through Turing's reaction-diffusion in one dimension. *a)* The diffusion rate of the inhibitor must be much faster than that of the activator. *b)* The patterning process is exemplified in a system of an activator (in green) and an inhibitor (in red). *i)* Initially, there is a quasi-homogeneous distribution of morphogens with some molecular fluctuations. In some places, the concentration of the activator might be high enough to produce auto-activation. *ii)* Auto-activation of the activator produces activation of the inhibitor. *iii)* As the inhibitor diffuses much faster than the activator, the activator is inhibited in the surroundings of the auto-activation area. This is known as lateral inhibition. *iv)* The change in molecular concentrations produces a chain reaction, hence making the activator auto-activate in areas where the inhibitor is not strong enough to repress it. *v)* A stable periodic pattern emerges. *Images i-v) modified from Green and Sharpe [2015]. Article is licensed under CC BY 3.0.*

real proof in biological systems was found during the years following its postulation. Furthemore, the fact that positional information was shown to be essential for patterning in *Drosophila* practically cosigned reaction-diffusion to oblivion [Green and Sharpe, 2015]. However, due to the advances in molecular biology and computational modelling, Turing's reaction-diffusion theory has been recently brought back to life. In fact, researchers have discovered biological systems with patterning regulated by reaction-diffusion. Examples are the development of palates of mammals, which Economou et al. [2012] discovered follow reaction-diffusion patterning in the form of periodic stripes, and the patterning of digits during limb development [Raspopovic et al., 2014]. Other studies have also proposed that reaction-diffusion plays a role in the distribution of hair follicles in mammals or feather buds in birds, germ layer specification and left-right patterning in early vertebrate development [Kondo and Miura, 2010; Marcon and Sharpe, 2012; Meinhardt, 2012; Roth, 2011] (as seen in [Green and Sharpe, 2015]).

Despite many authors including Wolpert positioning reaction-diffusion as the antithesis of positional information, the previous proofs of its existence in biological systems have made some authors consider whether reaction-diffusion and positional information can indeed complement each other and both take place in such biological systems [Green and Sharpe, 2015]. After all, both models deal with different mechanisms of morphogenesis.

#### 2.1.2.3 Combination of positional information and reaction-diffusion

In their hypothesis paper, Green and Sharpe [2015] describe three modes in which positional information and reaction-diffusion could be combined. Mode 1 (reaction-diffusion acting upstream of positional information) is based on the fact that in positional information, the way that the pre-pattern emerges is not specified. In fact, it could be produced by any mechanism able to produce patterning. Therefore, a reaction-diffusion system could be in principle the basis of such pre-pattern, which cells then use as positional information to interpret and decide their fate. As the authors say in their paper, "*the* [reaction-diffusion] *part describes the spontaneous self-organising manner in which the gradient has formed, and the* [positional information] *part describes the way in which cells subsequently interpret this gradient to choose different fates*". This means that both processes would be decoupled, i.e. working independently of each other, but complimentary. Müller et al. [2012] discovered that *Nodal* and *Lefty* proteins in zebrafish embryogenesis follow a reaction-diffusion model in the form of activator-inhibitor system. However, this system only has space to create half a wavelength, i.e. a peak-to-valley gradient, which is exactly what the positional information model uses. Even though it has not been formally verified, this could be an example of the mode 1 combination.

Mode 2 (reaction-diffusion acting in parallel with positional information) describes how cells could obtain information from both models independently but at the same time to shape tissue. As stated in §2.1.2.2, recent findings strongly suggest that a reaction-diffusion system defines the position where digits will be developed in mice [Raspopovic et al., 2014], i.e. reaction-diffusion establishes a pattern that differentiates digits from interdigits. In addition, morphogens gradients of ssh and bmp have been discovered to provide positional information for the formation of different types of digits in vertebrates [Tickle, 2006]. These findings seem to suggest that both positional information and reaction-diffusion could be working in parallel. For example, a pattern emerged from reaction-diffusion could specify the specific spatial locations of the digits, which then cells grow according to their interpretation of other morphogen gradients.

In mode 3 (reaction-diffusion acting downstream of positional information), the periodic wavelength pattern of a regular reaction-diffusion system could be altered by positional information. This means that the wavelength of the pattern would be modulated by the different interpretations to a morphogen gradient providing positional information. Even though a formal proof of this mode of operation has not been found yet, strong evidence for different wavelength patterns modulated by hox morphogens has also been found in mouse digit formation [Sheth et al., 2012]. In this example, the stripe-like pattern of digits is indeed a radial pattern spreading out from the limb bud. This means that the distance between digits near the wrist is shorter (a shorter wavelength) than that at the fingertips (longer wavelength). This could be caused by the modulation of the reaction-diffusion wavelength through the hox morphogen gradient created at the wrist of the limb.

The three modes are illustrated in figure 2.6.

### 2.1.3 A wider landscape of theories of development

The extraordinary diversity of organisms inhabiting our planet, each of them with common and distinct mechanisms of development as a result of evolution, exposes the difficulties of finding a unified theory of development that encompasses all of them, as Minelli and Pradeu [2014] argue in their book. Yet, this might not be desirable, because *"perhaps such a theory, if anything, would be an obstacle rather than an incentive to do actual research, as it would blinker people on other possible ways of seeing developmental processes"*, as these authors claim. Indeed, the field of developmental biology is being constantly updated, with new perspectives arising as new tools are developed and more experimental data is gathered and analysed computationally [Morelli et al., 2012]. Apart from the developmental mechanisms described in the sections above, other theories of development have been proposed. Current trends use information theory (for example, the free-energy principle [Friston, 2010]), the theory of computation or the theory of non-linear dynamical systems to understand developmental mechanisms, as Jaeger and Sharpe [2014] review. Bioelectricity in the form of long-term ion flows, voltage gradients and electric fields has also been recently found to complement molecular signals (i.e. morphogens) in the task of exchanging information among cells, tissues and organs during biological development [Levin, 2012]. This rich landscape of theories contributes to explaining development from different angles, each of them adding its own part to the bigger picture.

Among all the theories of development proposed so far, the classification of developmental

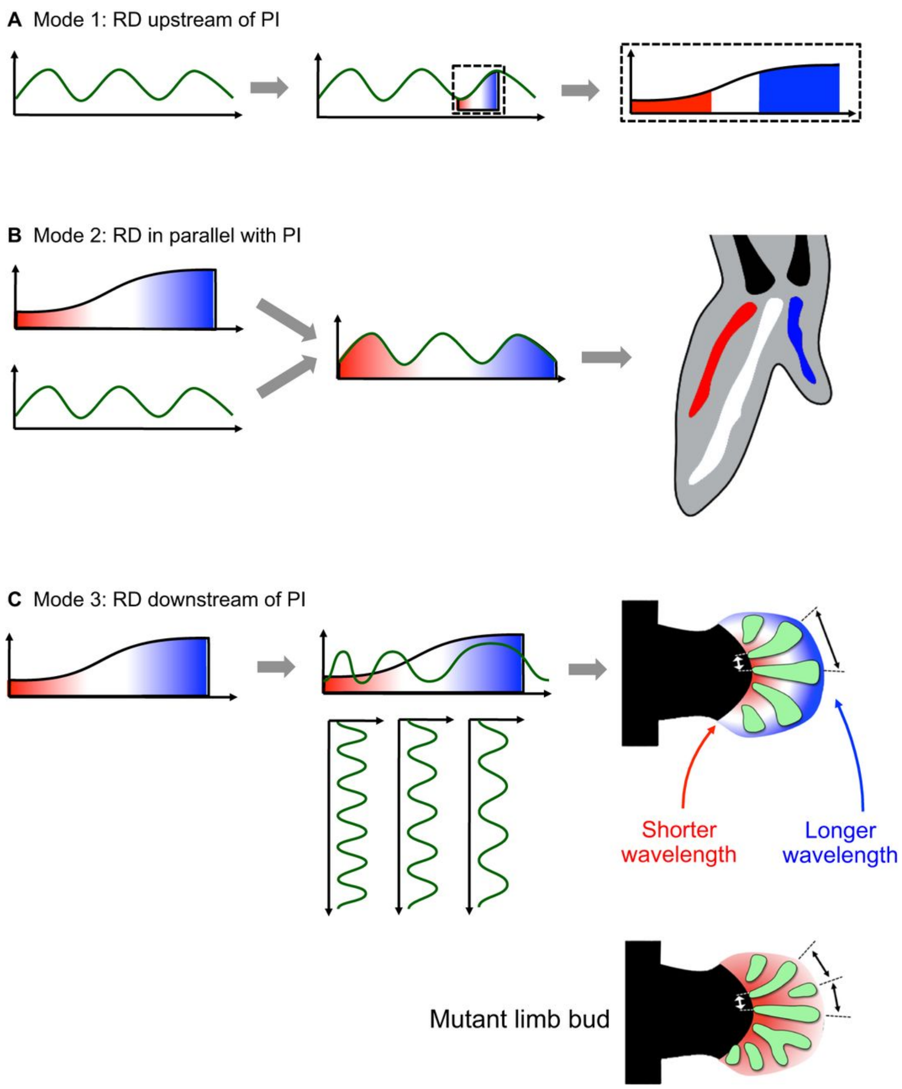


Figure 2.6: Hypotheses for the combination of positional information and reaction-diffusion. In mode 1, a pattern emerged from reaction-diffusion could serve as the pre-pattern for positional information to interpret, i.e. to define a pattern such as the French flag. In mode 2, reaction-diffusion could define the spatial pattern of digits in vertebrates, and positional information could define the specific type of digit to grow. In mode 3, different regions of the positional information gradient could be interpreted as different wavelengths for the reaction-diffusion pattern, hence provoking the splaying of digits in some vertebrates such as mice. *Image reproduced from Green and Sharpe [2015]. Article is licensed under CC BY 3.0.*

mechanisms proposed by Salazar-Ciudad et al. [2003] provides a simple classification that is good enough to take inspiration from as swarm roboticists. In fact, the purpose of chapters 3 and 4 is not to advance in the understanding of developmental biology, but to take inspiration from some of the developmental mechanisms proposed so far to design morphogenesis algorithms for robot swarms. In particular, the most interesting mechanisms described by Salazar-Ciudad et al. are the inductive mechanisms, because they are based on signals locally sent to neighbours (one of the core principles of swarm robotics), and the morphogenetic mechanisms of migration and adhesion, because they are directly related to motion. Therefore, these will constitute the basis of the algorithms developed in this thesis.

### 2.1.4 Quantification of morphologies: morphometrics

To quantitatively analyse shapes and understand morphological transformation in processes of growth and morphogenesis, biologists use morphometrics [Zelditch et al., 2012]. A more detailed explanation of them is given in the subsections below.

#### 2.1.4.1 Traditional morphometrics

Traditional morphometrics use parameters such as areas, lengths, angles and ratios among them. These are particularly useful when growth is to be quantified [Marcus, 1990]. For example, the perimeter of a shape, its area or the ratio between them could be used to measure form. However, these metrics alone do not give enough information about the shape. Furthermore, I was interested in dimensionless metrics to compare the shape of swarms with a different number of robots—either because of a decrease in the number of robots due to migrating robots becoming lost, or to compare experiments with a different initial number of robots.

More advanced metrics have been proposed, such as those comparing the shapes with respect to known shapes (e.g. circles, C-shape, etc.). In this respect, there are metrics such as the Circularity or Shape Index, both dimensionless. Circularity gives a measure of how much the convex hull (the smallest convex set that contains the shape) resembles a circle, whereas Shape Index gives a measure of how much the shape itself deviates from a circular shape—with $SI = 1$ for a circle, and $SI > 1$ for shapes different to a circle.

#### 2.1.4.2 Outline morphometrics

These methods approximate the contour of a shape with a mathematical function. Then, such function can be compared with the functions from other shapes in terms of coefficients, for example. Among these, particularly interesting is the use of Elliptical Fourier Analysis (EFA) [Kuhl and Giardina, 1982]. Contours can be approximated by combination of ellipses in the form of Fourier series. If the coefficients are normalised, the Fourier approximation is invariant to translation, rotation and scaling, hence making it suitable for a comparison among shapes

[Digumarti et al., 2019]. Algebraic complexity (AC) was defined by Ballaro et al. [2002] as the sum of all coefficients of the approximation of a shape through EFA. However, the drawback of this method lies in the approximation of the contour because a threshold must be used (e.g. approximation of the shape until the difference between perimeters is below a predefined value).

#### 2.1.4.3 Fractal morphometrics

A complex shape where the fine details are more important than the overall form can be seen as a fractal. Therefore, fractal analysis can be used to measure shape complexity. In this regard, metrics such as Fractal Dimension [Pennycuick and Kline, 1986; Annadhason, 2012] have been proposed to quantify this complexity. This metric gives a sense of how much the details of the shape change with respect to the scale at which they are looked at. Other metrics using this concept have been proposed to measure the complexity of the morphogenesis process itself—as opposed to only individual shapes. For example, the Double Log Fractal Dimension [Bissonette, 2012] uses a linear regression model between the logarithms of perimeter and area over time to characterise the complexity of growth. These metrics are particularly useful for detailed morphologies such as coastlines [Pennycuick and Kline, 1986] or plant leaves [Plotze et al., 2005].

#### 2.1.4.4 Information theory morphometrics

The process of self-assembly of individual parts to create a complex shape *"can be considered as the output of computational process"* [Soloveichik and Winfree, 2007]. Therefore, one could apply an information theory approach to characterise the computational program (e.g. a Turing machine) creating such shape, hence giving a sense of its complexity. In particular, this metric could be the Kolmogorov Complexity [Li and Vitányi, 2008], which is the length of the smallest program that produces the particular output (in this case, the shape being analysed). However, the Kolmogorov Complexity is not computable. Some approaches have been proposed in the literature to estimate the Kolmogorov Complexity based on entropy. In Chen and Sundaram [2005], the authors use entropy of the global distance distribution and entropy of the local angle distribution of points in the contour of the shape, as well as a randomness measure, to define a measure of shape complexity.

#### 2.1.4.5 Other morphometrics

In the field of landscape ecology, it is of great interest to quantify the heterogeneity of landscapes by identifying and analysing spatial homogeneous patches [Gustafson, 1998] to counteract the effects of human-induced biodiversity, as Moser et al. [2002] point out. In their work, they proposed a dimensionless metric of geometric complexity based on the contour of the patches, which can be useful here. This metric is called Number of Shape Characterising Points (NSCP), and it is defined as the minimum number of points required to define the contour of the shape.

The idea is that the greater the NSCP, the more complex the shape. This metric can potentially describe the morphology of the protrusions in robot swarms by means of the spikiness of shapes.

Other metrics have been proposed in the literature. For example, geometrical morphogenesis based on landmarks have been used the most by the community [Rohlf and Marcus, 1993; Zelditch et al., 2012] to compare the evolution of those landmarks over time. These have the advantage of being more powerful in statistically comparing morphologies because they capture more data.

Table 2.1 summarises all the metrics explained above.

## 2.2 Morphogenetic engineering

The wide array of spatially organising behaviours seen in nature has been a source of inspiration in robotics for a long time, especially for multi-robot systems. Apart from the impressive microscopic molecular behaviour of morphogenesis during embryo development described above, macroscopic morphogenetic behaviours can also be observed in examples such as ants building bridges to cross gaps [Anderson et al., 2002], termites constructing nests for protection from predators [Noirot and Darlington, 2000], or slime mould such as *Physarum polycephalum*, able to modify its shape to forage [Alvarado and Stephenson, 2017] (see figure 2.7). The existence of these examples certifies that some biological systems exhibit what is known as *self-organisation*, a process of organisation at the global level (the system) emerging from the local interaction of the components without any central controller leading the process or any global information about the whole state of the system [Camazine et al., 2003]. In addition, these examples also exhibit *architecture*, meaning that they spatially organise into structures with a certain purpose, this being the formation of a human body, an optimal path between a nest and a food source, or a mound. These properties turn these biological systems into complex systems with high levels of adaptability to changes in the conditions and robustness to perturbations. At the same time, some human-engineered systems are also self-organising, stochastic complex systems arising from rigid physical components, as Doursat et al. [2013] argue. For example, cities have emerged from buildings, internet from routers, or traffic jams from cars. Therefore, there are both natural complex systems that show architecture and artificial complex systems that show self-organisation. The possibility of convergence of both avenues was the catalyst for the recent creation of a new field of research: *morphogenetic engineering*.

Morphogenetic engineering seeks to design and implement systems that form functional, complex morphologies only through the local and decentralised interactions among the simple components that compose the system, with a special focus on controllability and programmability [Doursat et al., 2012]. There are two complementary but equivalent ways of achieving this. One is by introducing the principles of self-organised spatial organisation seen in natural systems into human-engineered systems (embedding informational systems in physics). Examples of this approach are programmable self-assembly in robotic systems [Rubenstein et al., 2014b]

Table 2.1: Morphometrics proposed in the literature for the quantitative analysis of shapes.

| **Name** | **Description** | **What it measures** | **How to calculate** |
|---|---|---|---|
| Area ($a$) | Area of the shape | Individual shape | Green's theorem (directly implemented in OpenCV) |
| Perimeter ($p$) | Perimeter of the shape | Individual shape | Summation of segments length |
| Perimeter/Area Ratio ($PAR$) | Relation between the perimeter and the area of the shape | Individual shape | $PAR = \frac{p}{a}$ |
| Shape Index ($SI$) | Deviation from a perfect circle | Individual shape | $SI = \frac{p}{2\sqrt{\pi \times a}}$ |
| Circularity ($CIR$) | Circularity of the convex hull | Individual shape | $CIR = \frac{4 \times \pi \times a}{p^2}$, where $p$ and $a$ are the perimeter and area of the convex hull |
| Fractal Dimension ($FD$) | Shape complexity across a range of spatial scales | Individual shape | Different methods apply (see Annadhason [2012]) |
| Double log fractal dimension ($DLFD$) | Complexity of the process in terms of evolution of shapes across a range of spatial scales | Whole morphogenesis process | $DLFD = 2 \times b$, where $b$ = slope of $FD$ for all shapes. The regression of $FD$ should be linear. If not, then do not calculate. |
| Number of Shape Characterising Points ($NSCP$) | The NSCP is an index characterising 2D shapes by the minimum number of points necessary to describe their contour | Individual shape | Algorithm described in Moser et al. [2002]. Pseudocode given in Algorithm 2. |
| Algebraic complexity ($AC$) | It is the degree (number of harmonics) of the Fourier series used to represent a contour | Individual shape | Fit a Fourier series to the contour until error in perimeter is below a certain threshold. Then, add up the values of the harmonics. |
| Shape complexity based on information theory ($SC$) | Measure of entropy to approximate Kolmogorov complexity | Individual shape | Algorithm described in Chen and Sundaram [2005] |

Figure 2.7: Examples of morphogenetic behaviours in nature. From left to right: ants building a bridge to cross a gap (*credit to Igor Chuxlancev*), several-meters-high mound built by termites (*credit to J Brew*), and the slime mould *Physarum polycephalum* foraging (*credit to Bernard Spragg*) *All images are licensed under Creative Commons.*.

or spatial computing [Zambonelli and Mamei, 2005], where a networked environment (e.g. a computer or sensor network) is augmented with a spatial layer that allows local communication using geographical information, hence bringing all the features of self-organised complex systems. Amorphous computing [Abelson et al., 2000; Nagpal and Mamei, 2004], where a massive number of copies of minimalist computing agents spatially and locally interact with each other, is another example of bio-inspired self-organisation into engineered systems. The other approach is by steering and embodying the self-organised behaviour of biological systems towards a more controlled and desired output (endowing physical systems with information). For example, the addition of termite-inspired pheromones to virtual wasps enhanced their computation abilities and improved the constructions that they could build in the work by Bullock et al. [2012].

In a nutshell, the aim of morphogenetic engineering is to achieve *programmable self-organisation* by creating self-organised engineered systems and/or programmable complex systems. The aim of this thesis is to take the route of embedding informational systems in physics to engineer a robotic system that performs a particular function for a real-world application with the high degree of adaptability to unforeseen/changing environments and self-repair after damage of biological systems. One way to achieve this is through a guided, bottom-up approach, whereby such engineered system is given a functional structure to develop as the end goal (a guided approach), but it finds its own way to produce it (a bottom-up approach).

The concept of morphogenetic engineering has been applied to synthetic biology. For example, Pascalie et al. [2016] proposed several shape-formation mechanisms for a simulated, abstract model of the bacteria E. coli. In their simulations, cells are able to perform developmental mechanisms such as division, molecule signalling/detection and apoptosis, driven by their internal genome (i.e. program). Starting from a single cell, the authors showed the ability of the system to grow differentiated cells to eventually develop regular and asymmetrical, limb-like morphologies. As the authors state, mechanisms like these could lead to a type of developmental 3D printing, useful for organ growth, biological computing or self-reconfigurable buildings. Tissue engineering was also explored by Doursat [2009]. In his work, embryogenesis was used as the inspiration for the design of simulated patterning and tissue growth in populations of cells by means of developmental mechanisms such as cell division, migration or adhesion. In the work by Nuñez et al. [2017], the authors proposed a computational model for bacterial colonies with a set of mechanisms (patterning and cell differentiation) that could be used as the building blocks to grant morphogenetic behaviour to the system. The authors applied their approach in simulation to show cellular functions such as metabolism, spatial gene expression and regulation.

Morphogenetic engineering is also particularly suitable for the field of swarm robotics, where a complex collective behaviour emerges from a large number of robots following simple rules and interacting locally in a distributed and decentralised manner [Şahin, 2005; Brambilla et al., 2013]. Many approaches based on the principles of morphogenetic engineering have been proposed so far for robot swarms. As this thesis mainly concerns the design of morphogenesis algorithms for large swarms of real robots, related work is reviewed in more detail in the next section.

### 2.2.1 Morphogenetic engineering for robot swarms

E. Şahin formally defined the field of swarm robotics in the mid 2000s. In his own words, *"swarm robotics is the study of how large number of relatively simple physically embodied agents can be designed such that a desired collective behaviour emerges from the local interactions among agents and between the agents and the environment"* [Şahin, 2005], as it occurs in swarms of biological systems such as ants, bees, termites, fish, birds, etc. What fundamentally distinguishes swarm robotics systems from multi-robot systems is how robots interact with themselves, and it is not related to the number of agents involved [Beni, 2005]. In multi-agent systems, individuals are able to perform the task on their own, but the completion of the task is enhanced by having several individuals collaborating towards it. On the contrary, robots in a swarm robotics system are unable to perform the task on their own, and therefore they need to collaborate. The result of these interactions is called *emergent behaviour* [Bonabeau et al., 1999; Şahin, 2005]: a global behaviour different from the individual's behaviour to accomplish a task at the swarm level. In addition, swarms in nature have been shown to be robust, scalable and flexible [Bonabeau et al., 1999; Camazine et al., 2003]. By taking inspiration from them, swarm robotics systems can potentially continue to operate despite perturbations in the environment or damage to individual robots (robustness), under different number of robots (scalability) and changing environments (flexibility, also known as adaptability) [Şahin, 2005]. However, these properties should always be tested and not taken for granted [Bjerknes and Winfield, 2013]. Therefore, the challenge of swarm robotics is to devise the individuals' rules so that the emergent global behaviour of the swarm shows the degree of adaptability, robustness and scalability seen in swarms of biological systems [Francesca et al., 2014].

Brambilla et al. [2013] classified robot swarm behaviours into four main categories: spatial organisation, navigation, decision-making and human-swarm interaction (HSI). In turn, each category is subdivided into more specific categories, which Schranz et al. [2020] recently extended. In particular, chapters 3 and 4 of this thesis fall into the topic of self-assembly and morphogenesis inside the category of spatial organisation, whereas work in chapter 5 falls into HSI. Below is a review of spatially organising robot swarms that have been proposed so far. A review of HSI work is given in §2.3.

Designing spatially-organising robot swarms could potentially lead to many applications in the real world [Brambilla et al., 2013]. Examples are self-constructing structures able to physically adapt to spatial conditions [Doursat et al., 2013; Werfel et al., 2014], reconfigurable modular robots able to modify their shape dynamically depending on the task (e.g. area coverage to clean gutters [Yim et al., 2007]), resilient swarm-based spacecrafts for space exploration [Vassev et al., 2012], enhanced urban search and rescue [Senanayake et al., 2016], or patterning at the micro/nano scale for biomedical applications [Hauert and Bhatia, 2014]. Many approaches have been proposed in the literature in the last 15 years, as reviewed by Oh et al. [2017]. I categorise them based on the methodology for creating shapes. It is worth highlighting that some approaches use a combination of different categories (e.g. morphology commands combined with light attraction towards robots). These will be categorised depending on the main essence of the strategy developing the morphology.

#### 2.2.1.1 Morphology commands

In this category, morphology commands are used to create a self-assembled robotic organism composed of individual swarm robots. Self-assembly in these systems is usually started from a robot seed that knows the shape to create, and invites other robots to join at the right location in its body. Then, information about the morphology is sent to joining robots for them to know how to invite other robots to join, eventually forming the desired morphology.

Mathews et al. [2017] proposed a self-assembly control paradigm inspired by animals' nervous systems whereby robots can merge into larger structures governed by one central unit named *brain unit*. In their approach, the units at the end of the structure gather information and transmit it to their parent units. The latter analyse and propagate the fused information up the structure until it reaches the brain unit. This unit then decides the actions to take and issues high-level commands which get propagated down the structure to the units, resembling the way a nervous system works. The advantage of their control methodology is that all robots retain their sensing and motion capabilities even when self-assembled. This allows them to split into separate bodies with their own brain units to form different morphologies if required, and detect and replace malfunctioning units, showing their ability to self-repair. The authors used the robotic platform marXbot [Bonani et al., 2010] to show how their approach could form different morphologies in experiments with up to 11 real robots.

The work proposed by Werfel et al. [2014] is a particular one, as it achieves external spatial organisation, i.e. robots build a 3D brick structure in a completely self-organised, distributed and decentralised fashion. By taking inspiration from the mound construction behaviour of termites, the authors proposed an algorithm for cooperative construction based on stigmergy, i.e. communication cues placed in the environment. In their work, stigmergy is provided by the presence or absence of bricks (the building blocks) in the structure, similarly to the presence or absence of soil in the termite mounds. A target 3D shape is pre-processed to determine the possible paths that robots can follow during construction, given their motion constraints (e.g. climbing up/down one-brick step at the most). After pre-processing, robots are given the set of paths. Robots always start navigating the structure from a seed brick. They continue navigating the rest of the structure, stochastically choosing among the allowed directions if more than one path is possible. Then, they place the brick in the appropriate location that satisfies a set of geometrical conditions (e.g. the brick is intended to be placed at that location), and navigate off the structure to load more bricks. The authors showed the constructing capability of their algorithm in simulation and in experiments with three real robots specially designed for their work. In addition, the authors showed how robots could repair the structure when part of it was manually removed.

As part of the REPLICATOR[2] and SYMBRION[3] projects [Kernbach et al., 2008], several self-assembly approaches were developed. These European projects sought to create evolvable, three-dimensional, symbiotic organisms composed of swarm robots able to self-assemble and self-disassemble for complex tasks. For example, Liu and Winfield [2012] showed in simulation different types of 2D organisms that could be created via self-assembly (with posterior disassembly) of the SYMBRION robots through a control methodology based on a recruitment-and-docking behaviour starting from a seed robot and local communication. The authors proposed two strategies for morphogenesis: one where all robots store the complete information about the organism to grow (global unique IDs are required), and another one where each robot only stores the information about the substructure to grow from itself. Information of the morphology is encoded in the form of a tree representation. Two years later, the same authors also showed that the same control methodology could grow morphologies with a heterogeneous swarm of robots [Liu and Winfield, 2014]. In their work, they developed three types of robotic modules that could self-assemble to create a SYMBRION organism: a *backbone* robot, able to move in the four cardinal directions on a flat surface, a *scout* robot, able to move on rough terrain, and an *activewheel* robot, able to carry and transport other robots. The authors showed the ability of

[2] http://www.replicators.eu/

[3] http://www.symbrion.eu/

simulated and up to 5 real robotic modules to self-assemble into a planar organism.

The SWARMANOID[4] project was another project that sought to create a heterogeneous swarm of self-assembling robots (a *swarmanoid*) [Dorigo et al., 2013]. For example, O'Grady et al. [2010] proposed different strategies for s-bots climbing a hill, crossing a hole or rescuing other robots as a single collaborative entity in a completely autonomous and distributed way. The swarm robotics platform called s-bot [Mondada et al., 2004] has the ability to physically connect to other s-bots (see figure 2.8). These robots were developed in the SWARM-BOTS European project [Dorigo et al., 2004]—SWARMANOID was the continuation of SWARM-BOTS. In the approach proposed by O'Grady et al. [2010], individual robots try to perform the task individually. If they detect they cannot do it alone, they self-assemble into a larger robotic organism from a robot seed—the first one that realises that a collective behaviour is needed. Robots emit light colour signals to attract/repel other robots. When self-assembled, they try to perform the task again. If they cannot do it (e.g. the hill is too steep), the seed robot triggers disassembly, and it commands a better morphology to create through self-assembly until the robotic organism can successfully complete the task. The authors performed experiments with up to 6 real s-bots. O'Grady et al. [2012] also showed the different morphologies (e.g. line, rectangle, square, star, arrow, shovel) that s-bots could achieve when physically assembled from a robot seed, and how these morphologies could be used to cross gaps, in experiments of up to 9 real s-bots and in simulation. In their approach, robots can create different morphologies based on *morphology-extension rules*, which determine where the next robot should physically self-assemble to with respect to the body of the robot inviting others to join the morphology. The advantage of this strategy is that the morphology to create is not stored in the robots—rules are reactive. However, the morphologies that robots can create are limited to periodically repeating morphologies, as the authors state. In their paper, another strategy to create arbitrary morphologies, named *SWARMORPH-script*, was presented. SWARMORPH-script combines the morphology-extension rules in the form of a program that robots can share with joining robots to know how to extend the structure appropriately to achieve a particular morphology. Li et al. [2016] proposed a similar approach, where swarms of Sambot robots [Wei et al., 2010] form 2D and 3D structures to overcome obstacles in the environment, as shown in simulation and with up to 9 real robots.

#### 2.2.1.2 Density-based systems

This category groups strategies based on robot cohesion (i.e. how close robots are to each other), swarm connectivity (i.e. how many neighbours robots have) or controlled dispersion, which are used for swarms to create different morphologies.

Nembrini and Winfield [2012] proposed completely decentralised control algorithms based on coherence for robot swarms performing area coverage, navigation towards a target, segregation and axis-formation. To achieve this, the authors came up with two strategies that they call

[4]http://www.swarmanoid.org/

Figure 2.8: A swarm of self-assembled s-bots navigating a rough terrain and climbing up a kerb. *Credit to Francesco Mondada and Michael Bonani*. Image is licensed under CC BY 3.0.

$\alpha - algorithm$ and $\beta - algorithm$. The $\alpha - algorithm$ tries to maintain at least $\alpha$ number of communication connections to other robots. The $\beta - algorithm$ uses the number of shared neighbours of a robot as the threshold for cohesion (i.e. to maintain at least $\beta$ number of communication connections), resulting in a better approach for maintaining swarm coherence compared with the $\alpha - algorithm$. Area coverage is achieved by changing the $\beta$ parameter. Swarm taxis is an emergent behaviour of the algorithm when sources of attraction (light attraction, i.e. phototaxis) are added to the environment, and some robots are attracted towards them. Segregation is achieved by having different $\beta$ parameters in the agents. Axis formation is achieved by considering different velocities to the robots performing phototaxis. If they move faster than the non-attracted robots, the swarm is elongated towards the source. If they move slower, they then grow perpendicularly to the source. Experiments were carried out in simulation and with up to 7 Linuxbots, which were developed at the Bristol Robotics Laboratory.

Hauert et al. [2009] applied the concept of maintaining connectivity to simulated flying robots creating a communication chain to connect two stations (e.g. a base station and a user station in a disaster scenario). To achieve this, the authors evolved the neural network controller of the robots, which only needed to know the heading of the robot, and the minimum number of communication hops from the robot to the base and user stations. Hence, their approach was completely decentralised, and used only local information obtained from neighbours (hop counts). Artificial evolution was used to optimise the parameters of the neural network controller. The fitness function (i.e. the metric to optimise during evolution) favoured systems that could quickly establish a connection between stations while maximising connectivity among the swarm robots. In simulation experiments, the authors characterised the robustness of their system to perturbations in the number of robots, communication ranges, time of initialisation and failures.

In the *SHAPEBUGS* algorithm proposed by Cheng et al. [2005], simulated robots move randomly in the environment until they encounter other robots. Then, using a local coordinate system constructed through trilateration, robots decide whether they belong to the shape or not—the shape to create is previously encoded as a map inside each robot. If they do, then a gas model is triggered, i.e. robots act as gas particles enclosed in a container, which is the shape to create. As a result, robots always keep a fixed distance with respect to other robots by moving towards areas within the shape with less density of robots. The advantage of this approach is that a shape is created independently of the number of robots that belong to it, providing there are enough robots. Self-reconfiguration is demonstrated in two experiments in simulation: one where the local coordinate system is perturbed, and another when part of the swarm is detached. In both examples the swarm was able to recover the shape. Self-repair was also tested in a scenario where part of the swarm suddenly disappears. By using the gas model, other agents could fill up the freed space of the shape to recover it.

#### 2.2.1.3 Attraction/repulsion

Work based on attraction/repulsion uses simple signals or gradients that are locally emitted from robots or objects in the environment to form robot morphologies. Such signals/gradients are generally transmitted through local communication or visual cues such as light. The difference with some of the approaches described in the category of morphology commands is that work here does not have a central robot with the information about the morphology to create, which then gets propagated across the swarm. Instead, shape formation occurs merely through the mechanism of attraction or repulsion to other robots or objects in the environment.

Liu et al. [2017] took inspiration from the searching and contraction behaviour of the unicellular slime mould *Physarum polycephalum* to develop a system of simulated agents that can find optimal solutions to spatial problems through morphogenesis. In particular, the authors showed how their approach could find the shortest path between two stations in a 2D grid, even in the presence of obstacles (e.g. a maze). In their work, two types of agents are defined. Both types can move forward in the grid, rotate around themselves, clone themselves or disappear from the grid. When moving, agents secrete chemicals to influence the behaviour of other agents, similar to trails of pheromones left by ants [Attygalle and Morgan, 1985] or the trail of mucus left by *Physarum polycephalum* [Reid et al., 2012]. The first type of agents, first appearing in the environment from one of the stations, is repelled by the chemicals left by the same type of agents, hence randomly expanding in the grid to search for the other station. When agents find the other station, they start to transform into the second type of agents. This time, the second type of agents attracts other agents through the chemicals (contraction). By combining the attracting behaviour of chemicals with the fact that an agent might disappear when it tries to move to an occupied grid cell, the shortest path between the two stations is eventually found. A similar approach was also developed by Jones [2011].

In Ferreira et al. [2018], the authors took inspiration from the self-repair capability of *Planaria*. Organisms of this species are able to regenerate completely when they are cut into pieces, obtaining as many new organisms as cuts [Reddien and Alvarado, 2004]. In their work, the authors simulated the regeneration process with two types of cells: neoblasts (able to divide) and somatic cells (not able to divide). Firstly, cells discover the morphology by tracking morphology messages that are transmitted from cell to cell. After this process, if a cell detects that a neighbour is missing, it then propagates a message that only neoblasts can respond to. This message attracts neoblasts to the location of the missing cells, first dividing to leave a cell in the original location, and then migrating towards the missing area. The authors showed in simulation how their mechanism could regrow most of the entire morphology with only a ratio of 10% of neoblasts.

In the work by Oh et al. [2018], the authors suggest three different algorithms for tracking and herding on swarms of simulated Kilobots and up to 23 real Kilobots. As described in §2.2.1.7 and §3.2, Kilobots are a low-cost platform for large-scale swarm robotics experiments [Rubenstein et al., 2014a]. The first algorithm is based on maintaining connectivity (it could be therefore categorised as a density-based algorithm), whereas the other two are based on local signals (morphogens) emitted from robots. In the second algorithm they propose, robots perform object tracking through gradient following, which means that robots move towards neighbours with a lower hop count with respect to the source of morphogen (the target). In the third algorithm, robots maintain the same morphogen gradient (i.e. the same initial formation) while tracking a moving target. If the hop count of a robot with respect to the source of the morphogen gradient changes, the robot follows other robots with lower/higher hop counts to restore its initial hop count. This work provided a demonstration that morphogens could be implemented in swarms of minimalist robots.

Özdemir et al. [2019] showed that modular robots can aggregate in a 2D grid by moving stochastically towards directions where they perceive other robots. In their approach, the authors proposed a completely decentralised *naive stochastic control policy* that robots use to decide where to move next with uniform probability, and using only 12 bits of information from their sensors—whether they perceive a neighbour or not in the four cardinal directions around the robots. The authors mathematically proved that their approach can almost surely reach a Pareto optimal configuration (the most optimal configuration for all conditions at once) in finite time. The authors also showed that the aggregation behaviour can be improved by having non-uniform probabilities when choosing the next direction of movement, which they optimised using artificial evolution. Furthermore, the authors showed that both strategies were scalable in simulations with up to 1000 robots, and that they could cope with noise levels of up to 50% in the sensory information. Finally, the authors implemented and showed their strategies with 6 M-Blocks [Romanishin et al., 2013]. Decentralised and distributed aggregation and line formation in M-Blocks were also shown by Romanishin et al. [2019].

Gauci et al. [2014b] proposed a particular type of aggregation algorithm for swarm robots with minimal capabilities, i.e. neither memory nor computation. In their approach, robots are attracted to other robots in the sense that they perform a specific motion behaviour when they detect neighbours with their binary sensor, which makes them move towards them. Such motion behaviour was found by exploring all possible types of motion controllers (with a discrete approximation), which were quadruple vectors with the velocity of each of the two motors for each of the binary outputs of the sensor. The optimum behaviour that the authors found corresponded to robots moving backwards in a circular trajectory until the detection of another robot, which triggers a rotation behaviour until the neighbour is no longer detected, going back to the backwards circular motion. The authors successfully showed the aggregating behaviour with swarms of up to 1000 simulated e-pucks [Mondada et al., 2009] and with 40 real e-pucks. Gauci et al. [2014a] also demonstrated the ability of this simple approach for the problem of object clustering. In their work, the authors optimised the motion controllers for this particular task given similar constraints. In particular, the only difference is that the sensor provides ternary information: whether there is nothing in front of the robot, or there is a robot or an object. The most suitable controller made robots move to the periphery of the objects and then move in a circular way to push the object inward. Experiments with up to 50 simulated e-pucks and 5 real e-pucks validated their approach. In the work by Özdemir et al. [2017], the authors extended the approach with 2 bits of information from the sensor to solve a shepherding task, showing successful results in simulation.

Bai et al. [2008] took an interesting approach whereby chemical fields emitted by agents in simulation were evolved to form the desired target shape. In their work, agents are attracted by the chemical fields released by other robots (chemotaxis), with the concentration of chemicals being directly related to the agents velocity. The advantage of their approach is that the desired morphology is created without agents knowing what shape is to be created or any leader, but as a response to the chemical attraction to other robots (i.e. local communication). However, the authors reported different levels of success in replicating shapes, from 4% to 100% depending on the particular morphology. Ellipse, diamond, hourglass, boomerang and gear-like shapes were tested.

Schmickl and Crailsheim [2007] took inspiration from the aggregating behaviour of the multicellular slime mould *Dictyostelium discoideum* for navigation of very simple agents in a cleaning task in simulation. The agents used only a binary signal (a light pulse) to simulate the chemical attractors producing chemotaxis that amoeba in *Dictyostelium discoideum* use to aggregate [Bonner and Savage, 1947; Alcantara and Monk, 1974]. In their approach, two stations representing the dirt area and the dump area emit different signals that are propagated across the swarm, in the form of waves, by robots. If robots are empty, they move towards robots closer to the dirt area, whereas if they are loaded with dirt, they move towards robots closer to the dump area. The authors showed that their methodology produced adaptable behaviours to environments with dynamic obstacles and the emergence of the shortest path between two sources. In addition, the authors showed that another strategy using a gradient signal as opposed to a binary signal, inspired by the release of the cAMP morphogen by the slime mould, could also be used for aggregation. The same authors also suggested another approach for shortest-path creation inspired by animal trophallaxis (the exchange of fluid food by direct mouth-to-mouth contact) in [Schmickl and Crailsheim, 2006]. The same cleaning scenario in simulation was considered. In this work, agents communicate their loading state and level of dirt load to neighbours, which helps the other agents move uphill (towards the dirt area) or downhill (towards the dump area), in a similar way as in the previous work.

As part of the SWARM-BOTS project, whose goal was to create a system (called *swarm-bot*) composed of physical self-assembling units to carry out tasks such as collective transport, navigation or exploration, Groß et al. [2006a] developed a self-assembly algorithm for s-bots to autonomously self-assemble, even if placed on rough terrain. In their work, robots self-assemble to a seed robot, dedicated objects representing a target, or other robots in the structure (but not seeds). For this, attraction/repulsion to specific colours displayed by robots is used to grow a robotic organism. As opposed to work described in the morphology commands category, this approach does not have any target morphology to create. The authors successfully showed the self-assembling capability of their approach in simulation and experiments with up to 16 real s-bots. In Groß et al. [2006b], the authors showed that the approach previously described could also be used for collective transportation of heavy objects towards a target location in experiments with up to 6 real s-bots.

#### 2.2.1.4 Reaction-diffusion

Turing's theory described in §2.1.2.2 has also been used by some authors to implement reaction-diffusion systems inside robots to drive pattern formation and morphogenesis at the swarm level. Reaction-diffusion in these examples is used for task differentiation or to influence motion.

Ikemoto et al. [2005] implemented a linear activator-inhibitor reaction-diffusion system in simulated and up to 12 real MK-01 robots for task differentiation, which could be useful for assigning different roles to some robots in the swarm. In their system, each robot has an internal representation of two molecules, $U$ and $V$, with $U$ being the activator and $V$ being the inhibitor. In their algorithm, robots first create a line formation using unidirectional infrared signals sent from the back of the robots. When other robots detect this signal, they move towards the robot emitting it, which is always the last robot that joined the line. Then, circle formation is executed. For this, the robot at the top of the line moves in an arc—while the rest follow it—, always maintaining the same direction of movement, until it finds the robot at the bottom of the line. When robots are in the circle, they execute the reaction-diffusion system. The reaction part is solved inside each robot (reaction happens between the molecules of a robot), and diffusion is calculated with the molecules received from the robots in its left and in its right hand side. The reaction-diffusion system produces a pattern whereby concentration of molecule $U$ is high every three robots in the circle, and low in the rest. As a consequence, robots with high concentration can differentiate with the rest—task differentiation. Differentiated robots can then move to create several spatial configurations such as triangular, quadrilateral or hexagonal morphologies. The authors also proved the ability of the swarm to recover the pattern when perturbed in simulation. This is an interesting work that shows how reaction-diffusion can be implemented in simple robots.

In Shen et al. [2004], the authors proposed a strategy based on a non-linear activator-inhibitor reaction-diffusion system of two molecules (that they call *hormones*) for a swarm of simple agents. They demonstrated in simulation that the swarm self-organised and performed behaviours such as search and seize of targets, area coverage, surmount and detour. To achieve this, robots and objects in the environment emit two molecules which influence the movement of others. In particular, the sum of both molecules is interpreted as an attraction or repulsion force for robots to choose the best direction of movement—they can sense bearing and distance to other neighbours. The authors showed how the swarm of simulated robots could recover after manually removing 15% of the robots—self-repair.

#### 2.2.1.5 Gene regulatory networks

As explained in §2.1.1, a GRN describes the complex interplay among different molecular regulators that control gene expression, mainly through proteins (which in turn affect the molecular regulators). Different GRN structures inside robots have been implemented inside robots, especially to influence motion.

In Meng et al. [2013], the authors use a GRN with genes secreting two proteins that directly affect the $x$ and $y$ direction of movement of the robot, respectively. The target shape is represented as a set of points, with as many points as robots. In their architecture, the target shape is included in the GRN for robots to be attracted to a unique point in the the shape to create. Robots converge to unique locations in the shape because they also diffuse one of the proteins to neighbours as a collision-avoidance mechanism. Hence, if a robot has already positioned itself in a point in the shape, other robots will move towards other points in the shape. Experiments in simulation showed that different 2D and 3D morphologies could be formed, as well as the ability of swarms to recover the shape after perturbation from a moving object and robot failures. However, their approach used a global coordinate system for robots to locate in the environment and move towards the shape. Guo et al. [2011] improved this approach with a local coordinate system constructed from a robot seed. They showed their results in simulation and with 8 real e-pucks. Jin et al. [2012] improved the system even further by means of a two-layer hierarchical GRN (H-GRN), where layer 1 generates a pattern depending of the target shape, and layer 2 controls $x$ and $y$ direction of the robot and collision avoidance. The difference with the work proposed by Meng et al. [2013] and Guo et al. [2011] is that the target shape is not predefined anymore. Instead, layer 1 of the H-GRN generates the pattern based on the position of the detected target that robots find in the environment. This way, their system can dynamically adapt to changing targets, while preserving the self-repair capability of the previous approaches. The authors showed that their H-GRN control system can be used to entrap several target objects in simulation and with 8 real e-pucks.

Taylor et al. [2007] also proposed two layers for controlling simulated underwater robots which can move horizontally and vertically. They combined a GRN with a cell-adhesion model (CAM), which describes how cells self-assemble to form aggregates in nature. The GRN controls the gene expression of several proteins that create bonds with other robots through virtual forces controlled by the CAM. The authors used artificial evolution to find the appropriate controllers for clustering and task differentiation in simulation. The combination of both the GRN and CAM layers produced more complex shapes and robust behaviour than each layer separately.

#### 2.2.1.6 Swarm Chemistry

Swarm chemistry is a model developed by Hiroki Sayama [Sayama, 2009, 2011, 2018] where simulated robot particles define spatiotemporal patterns only by interacting kinetically with each other—although there is information exchange between them. Particles have different sets of parameters defining their motion according to neighbours. They select the rule to use from their set of rules randomly with a certain probability, inspired by cell differentiation. When particles collide, they exchange their motion parameters depending on whether they are active or not. With a certain mutation rate, parameters can mutate during transmission to other particles or during their lifetime. Sayama was able to show that continuous stochastic re-differentiation (i.e. changing the motion rule at every timestep with certain probability) allowed the system to self-repair. For this, he performed two experiments where half of the swarm was removed and all the agents in a certain state of differentiation were removed. In the first experiment, the swarm regrew the shape that had created before removal with the remaining particles. In the second experiment, other robots differentiated to restore the missing state of differentiation in the swarm. Swarm chemistry is an interesting approach because it shows that morphogenesis can be achieved with minimal robot capabilities.

#### 2.2.1.7 Large-scale swarm robotics morphogenesis

A recurrent challenge in the field of swarm robotics has always been the demonstration of swarm behaviours with a large number of robots. Most previous work required precise motion or sensing abilities such as measuring angles to neighbours. In addition, overall validation of the shape-formation process itself, its robustness to perturbations and adaptability to changing environments was mainly done in simulation, or using few real robots (no more than a few dozens). Thus, it is unclear whether the self-organised shape formation algorithms proposed in the literature so far would cross the reality gap and scale up in a large swarm of simple, noisy robots.

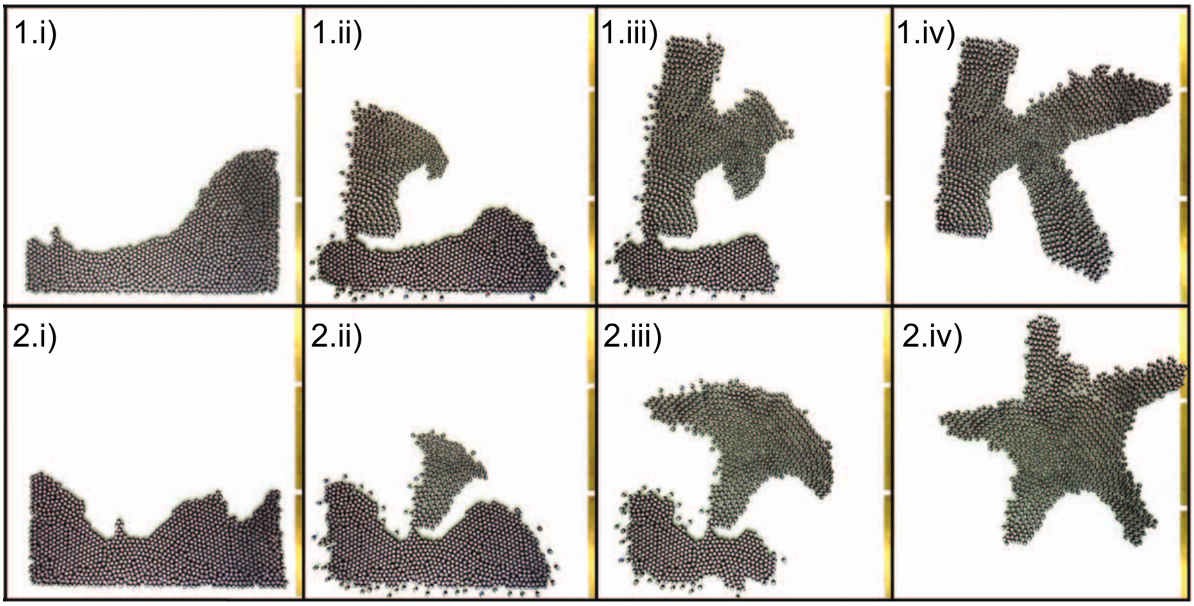


Figure 2.9: Swarm morphologies through self-assembly of Kilobots. *Image modified with permission from Rubenstein et al. [2014b]. © AAAS*

A significant breakthrough was made by Rubenstein et al. [2014a] when they created the Kilobot, a low-cost robotic platform especially designed for swarm robotics experiments. Rubenstein et al. [2014b] devised a decentralised, self-assembly algorithm with Kilobots. Swarms of up to 1024 Kilobots could successfully arrange themselves into pre-programmed morphologies such as a K-shape or a starfish shape (see figure 2.9). This was achieved with the help of a local coordinate system dynamically constructed by the Kilobots, starting from four preprogrammed seed robots. With it, each robot knew its relative position within the swarm despite only being able to communicate to their neighbours. Shapes were created by Kilobots filling the empty areas of the growing shapes by means of Kilobots moving around the edge of the swarm. Gauci et al. [2018] provided an alternative to this approach by programming the self-disassembly of the Kilobots to form a shape based on a local coordinate system as well. Kilobots started completely together as a square formation at the middle of their experimental arena. Then, robots used the coordinate system to locate themselves inside/outside the shape. Robots not belonging to the shape moved away from the swarm through light attraction/repulsion (phototaxis/antiphototaxis).

In the previous approaches, every robot had an explicit image of the final shape to create, hence depending on a top-down approach. In this respect, shapes were not fully emergent. From a morphogenetic engineering perspective, this places limitations on the flexibility and robustness of the system to unforeseen circumstances. As Davies [2013] states, self-assembly is not enough for the creation of large-scale, complex structures showing a high degree of flexibility. Instead, he advocates for the use of completely self-organised, bottom-up morphogenesis that incorporates the feedback systems seen in natural morphogenesis. In practical terms, this could be the morphodynamic mechanisms proposed by Salazar-Ciudad et al. [2003], where patterning affects morphogenesis, and vice versa. This idea could be useful for the development of a completely self-organised morphogenesis algorithm for robot swarms in chapter 3.

## 2.3 Robot swarms in society

Swarm robotics has been categorised as an underpinning technology with potential impact on all application areas of robotics, and it has been predicted to have considerable advances and impact during the decade 2020–2030 [Yang et al., 2018]. However, it is also generally impacted by negative public perception driven by science fiction and hyped headlines [Hamann, 2018]. To improve understanding and public trust of swarm robotics, it is important to engage with the public and relevant users/stakeholders to i) show examples of applications where robot swarms could be used as a tool to assist people, and ii) to identify societal requirements for robot swarms to become a reality. Indeed, the recent Special Eurobarometer on Attitudes Towards the Impact of Digitisation and Automation on Daily Life [Eurobarometer 460, 2017] showed that only one third of respondents felt comfortable with robots assisting them at work. The survey also found that people's opinion towards robotics and Artificial Intelligence was positively linked to the knowledge/awareness they have of such systems/applications. In the next two subsections, work related to engagement of users and the general public in the topic of swarm robotics will be reviewed.

### 2.3.1 User engagement

Within the field of robotics in general, abundant research on human-robot interaction (HRI) has been carried out in areas such as socially assistive robotics, industrial robots or search and rescue [Tsarouchi et al., 2016; Berg et al., 2019; Leite et al., 2013; Murphy, 2004]. In particular to multi-robot systems, one of the types of users that have mostly been approached by roboticists is personnel from disaster relief organisations (e.g. firefighters) [Delmerico et al., 2019]. Several user studies about search and rescue robots (not specifically swarms) can be found in the literature. For example, Driewer et al. [2005] used questionnaires and interviews to find out needs from fire brigades, governmental disaster relief organisations, nuclear power plants and military services. The authors found that the main user needs were i) live, layered maps to manually select the information (as opposed to having all the information displayed at once), ii) semi-autonomous robots for manipulation tasks, transportation of material, obstacle clearance, access to human-inaccessible areas and mapping, and iii) a reliable and robust communication system. In the study carried out by Yanco et al. [2006], a pre- and post-questionnaire methodology was used to gather information about users' experience with two prototypes of search and rescue robot systems. The authors found that visual information provided by the robots was crucial for users to accomplish the mission. Also, the authors suggested that semi-autonomy would be the best approach to leverage stress on users due to complete teleoperation of the robots.

More recently, Harbers et al. [2017] organised several workshops with firefighters to discover the most important aspects involved in search and rescue robotics, with a special focus on the ethical dilemmas arising from them. This study was part of the TRADR[5] project, which aimed to develop robots to assist human rescuers in disaster scenarios with a roboethics approach. From the workshops and interviews with users and stakeholders, the authors identified the following six ethical concerns/dilemmas:

1. Should robots be used even if they can also cause casualties?
2. Should research in search and rescue robots continue even if it could be used for military purposes?
3. Should robots be used even if they perform suboptimally compared to humans?
4. What is the right level of information processing to avoid loss/misinterpretation of data?
5. Should robots be deployed even if stakeholders might not understand them properly?
6. Should robots be deployed even if they could create responsibility-assignment issues?

With respect to swarm robotics, there has also been research on human-swarm interaction. For example, the GUARDIANS[6] project aimed to develop autonomous robot swarms to navigate and search warehouses on fire. Penders et al. [2011] describe their results with firefighters testing a smart helmet displaying information gathered by a swarm of robots (e.g. best direction of movement) in a test scenario. The authors observed lack of confidence from firefighters with respect to the guidance provided by the robot swarm, especially when they stopped their wall-following procedure to follow indications from the robots. Hence, user acceptance and trust is crucial for a successful deployment of technology.

In the study carried out by Podevijn et al. [2018] (see also [Podevijn et al., 2016a]), the authors measured the effect of the reality gap on humans' psychophysiological state, workload and responsiveness. The task was to supervise swarms of real and simulated e-pucks, with the latter consisting on 3D virtual reality and 2D computer simulations. A total of 37 participants with no previous robotics experience had to press a button whenever a robot in the swarm lighted up its LED in red. To quantify the reality-gap effect, the authors used subjective metrics in the form of questionnaires handed out to participants to measure pleasure/displeasure, mental alertness and workload. The authors also used objective metrics such as skin conductance levels to measure stress and reaction time to measure responsiveness to the swarm behaviour. The authors found that the real and simulated settings had different levels of effect on the participants, especially

[5] http://www.tradr-project.eu/
[6] https://vision.eng.shu.ac.uk/mmvl/viewfinder/guardians.html

between real robots (higher performance) and 2D computer simulation (lower performance). As a consequence, the authors suggested that using a virtual environment could mitigate the reality gap between simulation and real-robot experiments. In a similar study carried out by Podevijn et al. [2016b], the authors measured the impact of swarm size on humans. This time, a total of 24 participants were tasked with observing a swarm of 1, 3 and 24 real e-pucks moving randomly in the same bounded environment as the participants were. After analysing data from heart rates, skin conductance, mental alertness and pleasure/displeasure, the authors suggested that group size has a direct effect on humans, with an increase in size leading to an increase in the psychophysiological state, e.g. more stress.

Most of the rest of the work in human-swarm interaction has been done from a control perspective, meaning that researchers have focused on designing the control mechanisms to allow interaction between users and swarm robotics systems [St-Onge et al., 2019; Kolling et al., 2016; Nagi et al., 2014; Pourmehr et al., 2013; Nagi et al., 2012; Couture-Beil et al., 2010]. Thus, there is a lack of specific research on attitudes, concerns and requirements from potential users and stakeholders about the technology of swarm robotics. This is essential to successfully implement such systems in real-world applications, hence unleashing their economic and societal benefits [Winfield and Jirotka, 2018]. To address this situation, engagement of users on the topic of swarm robotics during the research and development process is needed.

The framework known as *mutual shaping* [Boczkowski, 1999] aims to precisely create a bidirectional relationship between users and technology researchers/developers. The objective is to incorporate societal choices in all stages of research and development by influencing both parties, hence moving technology away from a linear, deterministic development of technology *per se* (i.e. away from a because-we-can perspective). This approach facilitates the creation of *"more socially robust, responsive, and responsible robots"* [Šabanović, 2010]. Ethnographic studies, user-centered design or participatory design are high-level examples of mutual shaping methodologies [Lee et al., 2017]. Through ethnographic studies, researchers can understand users' behaviour towards a prototype of the technology—users are mere test subjects. In user-centered design, researchers usually find out about user needs, and use them to develop appropriate technology to cater for such needs—users are mere informers. A more power-balanced relationship between users and researchers is achieved through participatory design, where users are empowered to join the decision-making process of designing and developing technology. Not only do users inform researchers, but they also acquire knowledge about technology development, eventually applying it to the co-creation of the technology. Researchers also learn the meaning of technology from users, hence critically revising the technology. In fact, a recent survey on rescue robotics has stressed *"the importance of continued engagement with rescue stakeholders throughout the research process, to ensure that the priorities of both groups remain aligned"* [Delmerico et al., 2019]. Although mutual shaping in robotics has only been applied to the context of HRI and socially assistive robotics [Winkle et al., 2019], it could also be applied to swarm robotics.

### 2.3.2 Public engagement

Not only do users/stakeholders have to be included in the research and development process through mutual shaping, but the general public too, hence achieving a complete, societal co-development of the technology. By using mutual shaping tools (e.g. participatory design) and public engagement tools (e.g. community/partner engagement), swarm robotics in particular and any field of research in general can become science *for* society, whereby innovation is driven by a reflective, open, and democratic process, i.e. through responsible research and innovation [Owen et al., 2012; Stilgoe et al., 2013]. In turn, this could help establish the culture of *ethical governance* across research institutions and individuals [Winfield and Jirotka, 2018] to take into account ethical considerations from the beginning of the research process [Rainey and Goujon, 2011], resulting in an increase of public acceptance and trust of technologies. Because, all in all, *"technology is, in general, trusted if it brings benefits and is safe and well regulated"*, as Winfield and Jirotka [2018] state.

In the area of swarm robotics and multi-robot systems, public engagement has mostly been done from an educational perspective, i.e. teaching the general public about this particular technology. Public understanding of the technology is indeed fundamental to then have a more profound conversation about ethical aspects around it. Several educational robots have been designed to teach the wonders of collective behaviour. Some examples are eSwarBot [Couceiro et al., 2012], the Pi Swarm robot [Hilder et al., 2014], Pheeno [Wilson et al., 2016], HeRo [Rezeck et al., 2017] or Thymio [Mondada et al., 2017]. Among the low-cost alternatives, Thymio (version II) has the widest range of sensors and actuation capabilities, hence making it a good choice when the trade-off between functionality and cost is taken into consideration. It has been demonstrated in practical workshops that Thymio robots appeal to children and adults of both genders, that they are a suitable tool for different activities and coding skill levels, and that they make the user feel that they have learnt something new [Riedo et al., 2013].

Thymios were used in [Carrillo-Zapata et al., 2020] (a side study done during this doctorate), where a total of 9 workshop on bio-inspiration—with an explanation and demonstration of concepts such as emergent collective behaviour, linked to swarm robotics)—were delivered to primary and secondary school students in the French cities of Nancy and Metz. Our goal was to test whether students would still have fun, learn something new and gain an interest in STEM subjects (Science, Technology, Engineering and Maths), even when the workshops were conducted in a foreign language (English). After analysing the responses to questionnaires handed out to a total of 219 participants after the workshops, we found a direct correlation between having fun during the workshop and both learning something new and becoming more interested in STEM, as also stated in the literature of educational robotics [Karim et al., 2015]. In addition, our study suggests that there is a strong, direct correlation between the language being easily understood and the ability to have fun, and therefore learning and engaging in STEM subjects. Therefore, it is important to design educative experiences using a simple and fun approach.

Built around the concept of gamification, in the study carried out by Becker et al. [2014], the authors created an online game to teach the general public about manipulation of swarms, while giving insight to researchers about how participants best interacted with the swarms at the same time. They posed five different challenges to participants, where robots were disc-shaped and non-holonomic. Participants could choose to play as many of the five challenges as they liked. Over 3000 participants were registered. The first challenge consisted of an S-like maze where participants were tasked with the transportation of an object from one end of the maze to the other, with varying number of robots in the swarm (from 1 to 500 robots). Results showed that there was a local maximum in performance around 130 robots. The second challenge consisted of assembling a 3-block pyramid with a swarm of 16 robots using three different controllers: attractive (robots are attracted to each other), repulsive (robots are repulsed from each other) and a uniform global controller (robots move in the same global direction). Results showed that the attractive controller made players complete the task faster. The third challenge tested players' ability to solve the transportation task with varying levels of visual information about the swarm: with all the robots displayed, only the convex hull that surrounds them displayed, or only the mean (a dot) and the variance of the robot positions (an ellipse), or the mean alone. Results showed that using the minimum information, i.e. the mean, achieved the best performance. The fourth challenge applied varying levels of motion noise to the robots in the pyramid-assembly task, not greatly affecting performance. Finally, the fifth challenge investigated players' ability to create a particular morphology (letter A) with a different number of robots, from 1 to 10. Results showed that completion time increased linearly with respect to the number of robots used.

Even though the previous approaches are particularly good at increasing learning about the subject (in this case, basic concepts of swarm robotics), they only scratched the surface of all the possibilities for public engagement. In a similar way as it happens with users, there is a lack of understanding about how the general public perceives robot swarms, i.e. what the hopes and concerns about this technology are. For this, not only education is needed (from researchers to the public), but also a reflective process where the public is involved and influences research, hence educating researchers. In this respect, gamification has the potential to improve learning [Dicheva et al., 2015], and to encourage long-term reflections [Nicholson, 2015]. Indeed, long-term reflections could help establish a long-term engagement between the general public and researchers. Therefore, gamification could be an appropriate framework to build public engagement activities.

## 2.4 Concluding remarks

This chapter has provided the background on the main mechanisms for patterning and morphogenesis in multi-cellular organisms from a developmental biology perspective, as well as the state of the art in morphogenetic-engineering-based swarm robotics. Algorithms proposed in the literature so far have used a wide range of strategies, such as morphology commands, attraction/repulsion or morphogen concentrations, to achieve self-organised shape formation. However, when a very large number of robots has been used, robots were given a map of the shape to create, and swarms required preprogrammed seed robots and the creation of a coordinate system, hence not representing an example of completely self-organised morphogenesis, as seen in biological systems. Self-organised morphogenesis without the previous requirements has been validated only in simulation or with a very few real robots. As a result, there is a gap in demonstrating completely self-organised, bottom-up morphogenesis in large swarms of real robots without using any map, preprogrammed seed robots or coordinate systems. To address this gap, chapters 3 and 4 build on previous work proposed in the literature, and take inspiration from some of the developmental mechanisms described in the first section of this chapter (e.g. morphogen gradients through simple morphogen diffusion, patterning based on reaction-diffusion models, or cell migration), to fulfil the aim of completely self-organised, controllable and functional morphogenesis in large swarms of real robots. Concepts from the swarm robotics literature such as morphogenesis using minimal robot capabilities, shortest-path creation between two stations and connectivity maintenance across the swarm could be useful for adding controllability and functionality to the robot swarms.

In addition, a gap in user and public engagement with respect to attitudes, concerns and requirements on the topic of swarm robotics has been identified in this chapter. If morphogenetic robot swarms are to be successfully deployed for real-world applications, this is a crucial aspect to tackle. In chapter 5, this gap is addressed by considering the inclusion of users and the general public in current swarm robotics research. The framework of mutual shaping will be used to study attitudes, concerns and requirements from fire brigades, which are potential users of the technology developed in chapters 3 and 4. The framework of gamification will be used to study hopes and fears from the general public. With these two studies, I contribute to shaping the future development of robot swarms; a future where advances in this technology are mutually beneficial for researchers and society.

# Chapter 3

## Bio-inspired morphogenesis in robot swarms

Self-organised shape formation with large numbers of robots could allow for applications in areas such as search and rescue, utility inspection/maintenance, disaster relief, architecture, construction, space engineering or biomedicine. The ability of robot swarm to create spatial shapes in a completely emergent fashion, without a pre-established morphology, but adaptable to environment conditions and robust to perturbations, could be crucial for this type of applications. In this chapter, a completely self-organised, bottom-up morphogenesis algorithm for large swarms of real, simple robots is presented. Swarms do not need to use any map of the shape, coordinate system or preprogrammed seed robots. Instead, inspiration comes from the self-organised morphogenesis principles seen in nature to endow them with morphogenetic behaviour. In particular, from inductive and morphogenetic developmental mechanisms such as patterning based on a reaction-diffusion system, which in combination with robot migration produces emergent, adaptable and robust morphogenesis.

This chapter begins by summarising relevant work that our morphogenesis approach builds on, followed by sections describing the technicalities of the algorithm, and results of a total of 121 simulations with swarms of 1000 robots and 20 experiments with swarms of 300 Kilobots. Then, I discuss results in terms of emergence of regular and organic-like shapes, adaptability to different initial configurations (circular and rectangular), and robustness to minor damage (cutting off protrusions) and major damage (splitting swarm in half).

The work presented in this chapter is the result of a collaboration with the European Molecular Biology Laboratory led by James Sharpe in Barcelona, Spain (previously known as the Centre for Genomic Regulation). James Sharpe's team study reaction-diffusion in embryogenesis. Much of the written content here presented has been published in the following peer-reviewed journal:

- Slavkov, I.*, Carrillo-Zapata, D.*, Carranza, N., Diego, X., Jansson, F., Kaandorp, J., Hauert, S., & Sharpe, J. (2018). Morphogenesis in robot swarms. Science Robotics, 3(25), eaau9178.

– *: co-first authors.
– **Personal contribution**: I performed the large-scale morphogenesis experiments with Kilobots, as well as the parameters exploration in simulation. I also performed the quantitative analysis of emergence, adaptability and robustness to damage, and analysed the experiments with respect to morphometrics. Finally, I contributed to the writing of the paper as co-first author.

Some parts of §3.3.3 and §2.2 have been reproduced verbatim from the publication above.

## 3.1 Introduction

Granting human-engineered systems the ability to create controllable and functional spatial structures composed of multiple components with similar levels of emergence and self-repair as the ones seen in biological systems is one of the aims of morphogenetic engineering [Doursat et al., 2013]. In essence, the field of morphogenetic engineering seeks to design and implement programmable self-organisation leading to spatially-organising systems that can cope with changing environments (adaptability) and perturbations (robustness). This framework could be highly beneficial for buildings with the ability to repair cracks or adapt to strong weather conditions, robot swarms exploring a disaster environment, or swarms of nanoparticles penetrating through tissue to deliver drugs.

As described in chapter 2, many approaches have been proposed for the problem of spatial organisation in robot swarms. In particular, it has been shown that self-organised morphogenesis can be achieved with minimal robot capabilities [Özdemir et al., 2017; Gauci et al., 2014a,b; Sayama, 2009; Schmickl and Crailsheim, 2007; Schmickl and Crailsheim, 2006], that swarms can adapt to changing environments and recover from damage [Ferreira et al., 2018; Mathews et al., 2017; Werfel et al., 2014; Hauert et al., 2009; Cheng et al., 2005], and that very large swarms of robots can self-organise into different morphologies [Gauci et al., 2018; Rubenstein et al., 2014b]. However, self-organised morphogenesis with a large number of real robots has only been demonstrated using a map of the desired shape, preprogrammed seed robots and a coordinate system. On the other hand, morphogenesis without using any map, preprogrammed robots or coordinate system has only been shown in simulation or with few real robots. Given the potential applications of spatially-organising robot swarms, it is necessary to devise completely self-organised morphogenesis algorithms that can be shown to be emergent, adaptable and robust in large swarms of stochastic robots.

This chapter poses this particular question: can large swarms of real robots form fully self-organised shapes that are adaptable and robust? To address it, we take inspiration from multi-cellular embryogenesis, where patterning and morphogenesis play a crucial role in shaping and growing tissue, as described in §2.1. In this chapter, I show how some of the inductive and morphogenetic developmental mechanisms seen in biological systems can be combined to give rise to emergent, adaptable and robust shape formation in swarms of 300 Kilobots. Concretely, we use the spontaneous symmetry-breaking phenomenon of reaction-diffusion systems found in some examples of multicellular tissue development [Raspopovic et al., 2014; Scholes et al., 2019] (as described in §2.1.2.2), as well as migration, which has been shown to play a crucial role in several developmental processes [Keller, 2005] (as described in §2.1.1). The existence of feedback loops in these processes make them robust to noise [Salazar-Ciudad et al., 2003; Salazar-Ciudad and Jernvall, 2004], which could be useful for the engineering of completely self-organised shape formation in robot swarms. Therefore, these mechanisms are the main building blocks of the morphogenesis algorithm described in this chapter.

Some researchers have also used inductive and morphogenetic developmental mechanisms to achieve self-organised shape formation. As described in chapter 2, inductive mechanisms have been used in the form of morphogen gradients, reaction-diffusion systems and gene regulatory networks [Oh et al., 2018; Meng et al., 2013; Jin et al., 2012; Guo et al., 2011; Ikemoto et al., 2005; Shen et al., 2004]. The core idea behind the previous examples from the literature is that robots represent cells that secrete molecules affecting their motion depending on the particular concentration of such molecules. This concept seems very useful due to the fact that robots react only to such concentrations to self-organise into different morphologies without using any map of the shape or coordinate system (in some of them). They also show a high degree of adaptability and robustness to perturbations.

In biological systems, the spatial and non-uniform molecular patterns emerged from reaction-diffusion eventually drive a secondary process of cell differentiation, which might result in coordinated cell movement (migration), tissue proliferation (cell replication) or apoptosis (cell death). Among the previous cell fates, cell migration is particularly interesting for our problem because it can be implemented in robots through movement. For example, this was shown by Rubenstein et al. [2014b] and Gauci et al. [2018]. In their work, shapes were constructed by robots migrating to areas within/outside the shape (with the help of a map and a local coordinate system). What is particularly interesting in their work is the idea of the edge-following behaviour, whereby robots always maintain connectivity with their neighbours and perform shape formation at the same time. This could be useful in scenarios where maintaining connectivity is crucial (e.g. swarm-guided navigation [Brambilla et al., 2013] or search and rescue [Murphy et al., 2008]).

All the previous examples show that it is possible to use such inductive and morphogenetic mechanisms in swarms of simple robots. By combining them, completely self-organised shape formation could be achieved. This combination could be interpreted as localised tissue growth in some areas of the pattern, and localised cell death others, as seen during tetrapods' digit formation [Wolpert et al., 2015]. We propose to use an activator-inhibitor reaction-diffusion system combined with edge-following robot migration. Many models of reaction-diffusion systems for two molecules have been proposed since Alan Turing's theory of morphogenesis [Murray, 1982]. From linear systems such as the one originally described by Turing [1952], to non-linear systems such as the ones proposed by Gierer and Meinhardt [1972], Thomas and Kernevez [1976], Schnakenberg [1979], or Gray and Scott [1984]. We chose to use a reaction-diffusion system with no conservation of mass, i.e. the sum of concentrations of molecules changes over time, for simplicity reasons. The simpler the model, the more suitable it is for simple robots such as Kilobots. This is appropriate to overcome the presence of noise, hence providing a better solution for stochastic robot swarms. The model we use is the originally described by Alan Turing, which Miyazawa et al. [2010] showed in simulation that it can produce several spatial patterns by varying two of its parameters.

In summary, in this chapter I show how the ideas of robots representing cells that produce a molecular pattern based on reaction-diffusion, and edge-following movement around the swarm to migrate to areas of growth defined by the pattern, can be combined. By merging both processes in a large-scale feedback loop wherein both occur and drive each other simultaneously (except when the pattern is not stable at the beginning, which is when migration is not taking place), dynamic morphogenesis is achieved, hence increasing adaptability and robustness to damage. Throughout this chapter, I also show how the morphogenesis algorithm here presented produces emergent morphologies, even when swarms start from different initial configurations, and how swarms manage to recover from damage by regrowing missing parts or rejoining the swarm.

## 3.2 Experimental setup

The chosen robotic platform to demonstrate completely self-organised morphogenesis in real robots is the Kilobot [Rubenstein et al., 2014a]. Kilobots are minimal, low-cost robots designed to enable swarm experiments in large numbers, as explained in §2.2.1.7. Indeed, it is the only swarm robotics platform that has been used for experiments reaching the three figures in the number of robots used (see table 1 in [Schranz et al., 2020]). Kilobots are equipped with two vibrating motors on each side, infrared communication up to 10cm, an ambient light sensor and an LED to communicate their internal state (see figure 3.1). Kilobots run code autonomously after uploading it through infrared signals sent from a controller that is connected to a computer. Even though Kilobots lack precise motion and sensing due to their low-cost design, they are an ideal platform to demonstrate large-scale collective behaviours. That is the reason for choosing them for the experiments in this chapter and chapter 4.

A simulator was used to test and refine the algorithms developed in this thesis before taking them to the Kilobots. The chosen simulator was Kilombo [Jansson et al., 2015], as it was specifically designed to simulate Kilobots. Other Kilobot simulators have been proposed (e.g. V-REP with the Kilobot module, or KBsim). The advantage of Kilombo is that it is written in C language. Therefore, the simulation is compiled (i.e. as a standalone program), as opposed to

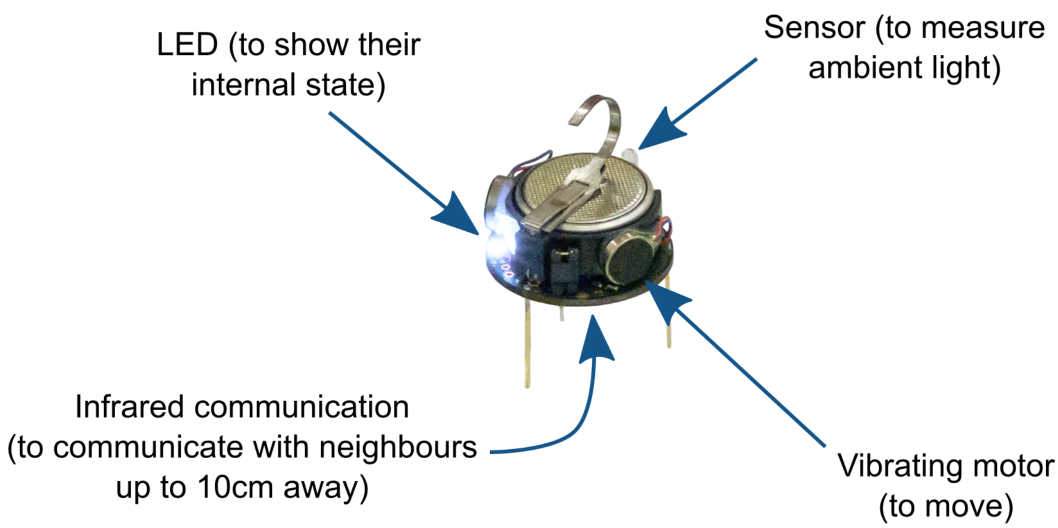


Figure 3.1: Description of a Kilobot [Rubenstein et al., 2014a].

interpreted (i.e. where instructions are executed in real-time), as in the case of other simulators. This allows for optimisation, hence being able to run much faster simulations with a large number of robots. Indeed, the developers showed the ability of Kilombo to simulate swarms of 1000 Kilobots 100 times faster than real time, which makes it a suitable platform for testing the algorithms developed here.

Finally, real-robot experiments were carried out in the swarm robotics arena available at the Bristol Robotics Laboratory. This arena consists of a 3x2m flat surface where Kilobots are positioned. The surface is completely reflective, so infrared messages can be sent from Kilobot to Kilobot—they send infrared pulses downwards in the form of a cone that bounce and get reflected on the surface for other Kilobots to pick up. Above the arena there is a high-definition camera to record experiments.

## 3.3 Methodology

The morphogenesis algorithm developed in this chapter mainly consists of two parts: patterning and migrations. At the beginning, robots in a static swarm are initialised with random concentrations of two molecules, $U$ and $V$—more details given in §3.3.1. Then, a spots-like pattern composed of robots with high concentration of molecule $U$ (robots in green, blue and purple) emerges in the swarm through reaction-diffusion (also known as *Turing pattern*). For convenience in this chapter, we call these spots *Turing spots*. Robots at the edge of the swarm and in low areas of concentration of molecule $U$ (robots in grey) start migrating to areas of high concentration of molecule $U$ by orbiting the swarm until they find such robots. In parallel with robot migration, the pattern adjusts to the new morphology, i.e. molecules concentration in both the newly positioned robots and the robots that had already become part of the Turing spots changes. This makes the

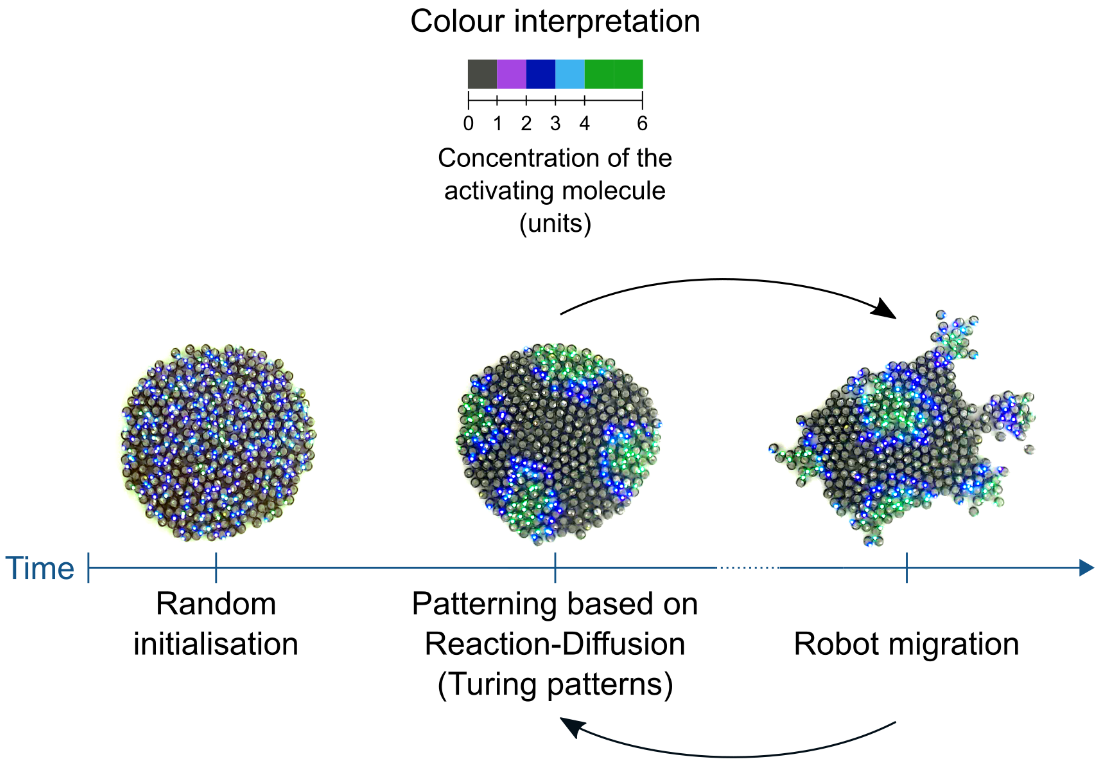


Figure 3.2: Overview of the steps resulting from the bio-inspired morphogenesis algorithm. Random initialisation of molecules $U$ and $V$ is followed by a process of patterning and robot migration. Patterning and migration run alongside of each other to achieve dynamic morphogenesis.

pattern shift towards the edge of the swarm, hence making other migrating robots stop by them. This process of continuous patterning and migration produces emergent shapes in the form of protrusions. An overview of the morphogenesis algorithm is shown in figure 3.2. Figure 3.3 shows a detailed example of the execution of the morphogenesis algorithm, as explained before. A more detailed description of both patterning and migration is given below.

### 3.3.1 Pattern formation

As described in §2.1.2.2, reaction-diffusion is a mathematical model which describes the process whereby certain molecules (named *morphogens*) chemically react with each other and diffuse through space. Under the right conditions, they self-organise into structured patterns from pure random initial concentrations. As the goal is to implement a reaction-diffusion system inside simple robots, we chose the simplest system for this, i.e. the one requiring less computation time and memory. We implemented a simple, linear reaction-diffusion system in the form of an activator-inhibitor network of two molecules, $U$ being the activator and $V$ the inhibitor. This linear model was also used by Miyazawa et al. [2010]. In this network, molecule $U$ activates the production of itself and molecule $V$, whereas molecule $V$ inhibits molecule $U$, as shown in

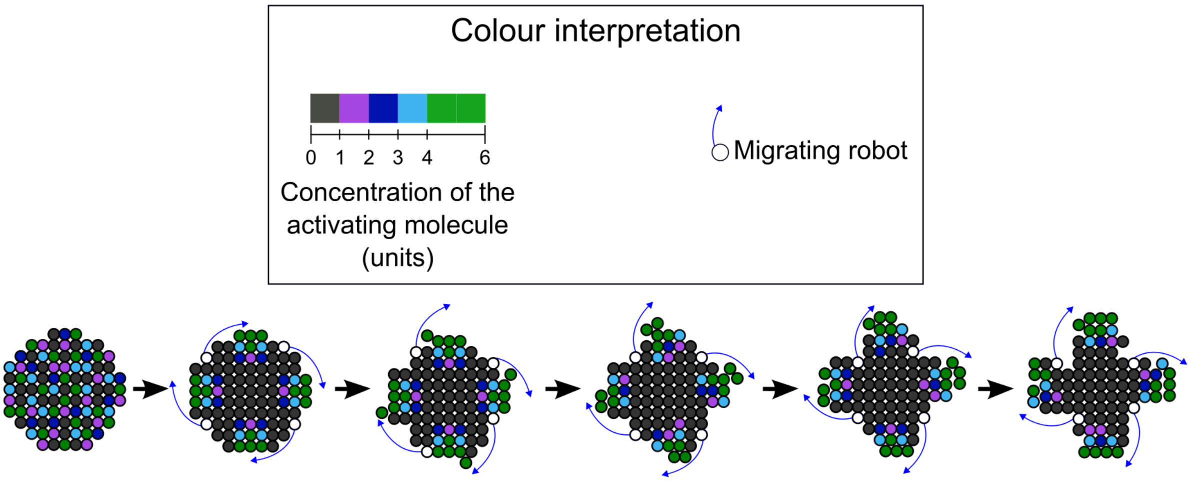


Figure 3.3: Conceptual execution of the bio-inspired morphogenesis algorithm. Robots are initialised with random concentrations of molecules $U$ and $V$. Then, a pattern emerges in the swarm through reaction-diffusion of both molecules. Robots in areas with low concentration of molecule $U$ orbit the swarm and stop by the proximity of areas with high molecule concentration (migration). The pattern adjusts to the new morphologies, making the Turing spots be at the edge of the swarm. This morphogenetic process of patterning and migration makes the swarm grow protrusions.

figure 3.4. The rate of change in the concentration of molecule $U$ ($\Delta u$) and molecule $V$ ($\Delta v$) is governed by the following linearly coupled equations encoded in the robots:

$$\frac{\Delta u}{\Delta t} = Rf(u,v) + D_u \nabla^2 u \tag{3.1}$$

$$\frac{\Delta v}{\Delta t} = Rg(u,v) + D_v \nabla^2 v \tag{3.2}$$

Equations 3.1 and 3.2 consist of two parts: the reaction and the diffusion part. In these equations, $R$ is the reaction parameter (to ease parameters adjustment) and $D_u$ and $D_v$ are the diffusion parameters. The production and destruction of molecules $U$ and $V$, also known as the reaction kinetics, is governed by the following linear functions $f$ (for molecule $U$) and $g$ (for molecule $V$):

$$f(u,v) = (Au - Bv + C) - \gamma_u u \tag{3.3}$$

$$g(u,v) = (Eu - F) - \gamma_v v \tag{3.4}$$

Parameters $A,B,C,E$ and $F$ in equations 3.3 and 3.4 are directly related to the number of molecules that are produced/destroyed. Parameters $\gamma_u$ and $\gamma_v$ represent the degradation of both molecules. The maximum number of molecules that are synthesised is limited by the parameters $synU_{max}$ and $synV_{max}$, with $0 \le (Au - Bv + C) \le synU_{max}$ in function $f$, and $0 \le (Eu - F) \le synV_{max}$

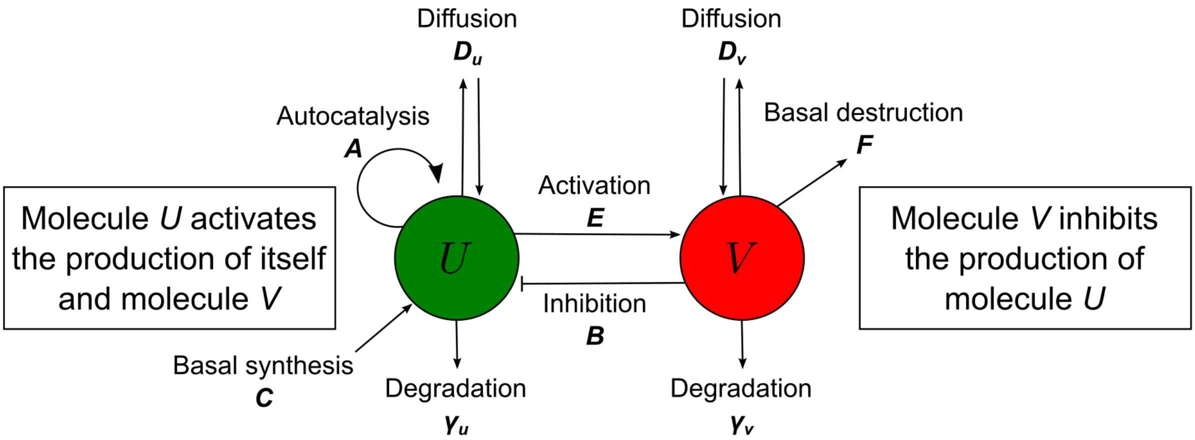


Figure 3.4: Activator-inhibitor network implemented in the Kilobots for the reaction part of the reaction-diffusion system, plus description of each parameter.

in function $g$. With these limits, different levels of concentration can be defined (e.g. to distinguish between low and high concentration of molecules). To update the system, a discrete approximation for the differential equations 3.1 and 3.2 with a timestep $\Delta t$ was used. Finally, $\nabla^2 u$ and $\nabla^2 v$ are the Laplace operators for the diffusion part of equations 3.1 and 3.2, with $n$ being the number of neighbours of the robot and $u_j$ or $v_j$ being the corresponding molecule concentration of neighbour $j$:

$$\nabla^2 u = \sum_{j=1}^{n} (u_j - u) \tag{3.5}$$

$$\nabla^2 v = \sum_{j=1}^{n} (v_j - v) \tag{3.6}$$

Each robot is programmed with an internal representation of both molecules $U$ and $V$. At the beginning, robots are initialised with random concentrations chosen from a uniform distribution in the range 0 to 6 units (this range was selected after experimentation in simulation). Then, at every timestep, the molecules inside the robots react with each other, increasing or decreasing their concentration $u$ and $v$ by activating (i.e. producing) and inhibiting (i.e. destroying) themselves. Diffusion is approximated through message passing by sending and receiving molecules from their neighbours. Robots sum up the differences between the molecules concentrations of their neighbours within 85mm (from manual experiments, a greater communication range highly increases the probability of losing messages) and their own molecules concentration to implement diffusion, as shown in figure 3.5. For that, they send the following information to neighbours: their locally, unique ID (2 bytes), their running average of neighbours (1 byte), the state the robot is in (1 byte), their concentration of molecule $U$ (2 bytes) and their concentration of molecule $V$ (2 bytes). Each robot then stores the information received from neighbours in a table. They also compute and store a running average of neighbours of their neighbours (useful for edge detection,

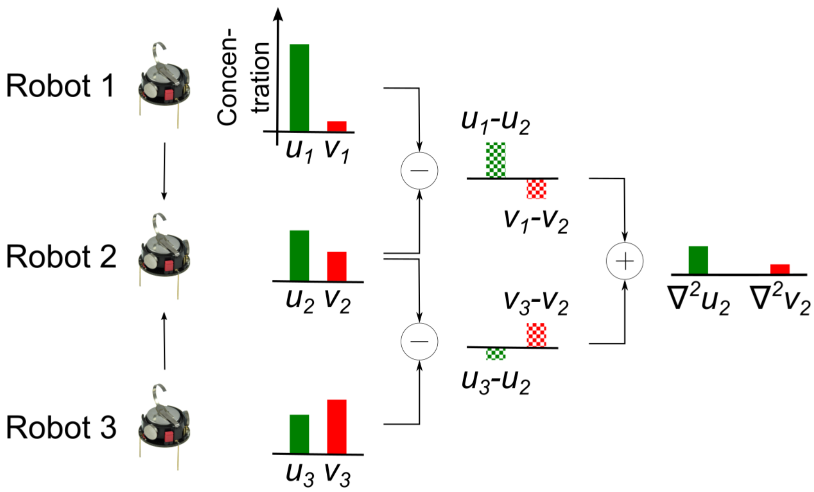


Figure 3.5: Molecules diffusion (in this example, for robot 2) is calculated by subtracting neighbours' concentration of molecules from the concentration of molecules in a robot, and summing the differences up.

as explained in §3.3.2), the distance to each neighbour and a timestamp of when the last message from this neighbour was received (to remove entries older than 2 seconds to keep information up to date).

As shown by Miyazawa et al. [2010], different combinations of parameters $A$ (autocatalysis of molecule $U$) and $C$ (basal synthesis of molecule $U$) produce different patterns, from spots to stripes and inverted spots (see figure 3.6). For our morphogenesis algorithm, we used the same coefficients for all the parameters as in Miyazawa *et al.*'s work, except for $R$, which we had to adapt (in simulation) to match their spot-like patterns—$R = 20$ in their experiments, whereas $R = 160$ in ours. In fact, parameter $R$ is only a scaling factor, hence, the system conserves the proportion between the values of the parameters. Spots provided an adequate compromise between the number of robots in areas of low concentration (migrating robots) and robots in areas of growth (with high concentration). If other patterns (for example, stripes or inverted spots) were used, there would not be enough robots to move at the edge of the swarm, thus negatively affecting migration. The coefficients that produced spots in our robot experiments were: $R = 160$, $D_u = 0.5$, $D_v = 10$, $A = 0.08$, $B = 0.08$, $C = 0.03$, $E = 0.1$, $F = 0.12$, $\gamma_u = 0.03$, $\gamma_v = 0.06$, $synU_{max} = 0.23$, $synV_{max} = 0.5$, $\Delta t = 5 \times 10^{-5}$. These values are exactly the same as given in Mizayawa *et al.*'s work (except parameter $R$, as explained above). The pattern can be visualised by having different colours depending on the concentration of the activating molecule $U$. The LED of the Kilobots was set to the following colours: green LED for $u > 4.0$, teal for $3.0 < u \leq 4.0$, blue for $2.0 < u \leq 3.0$, purple for $1.0 < u \leq 2.0$, and off for $u \leq 1.0$.

After approximately 10 minutes in real time, a stable pattern emerges in the swarm. Then, robot movement starts. A hard-coded timer *WAITING_TIME* of 10 minutes was set in the robots to start motion. At this point, patterning and migration occur alongside of each other.

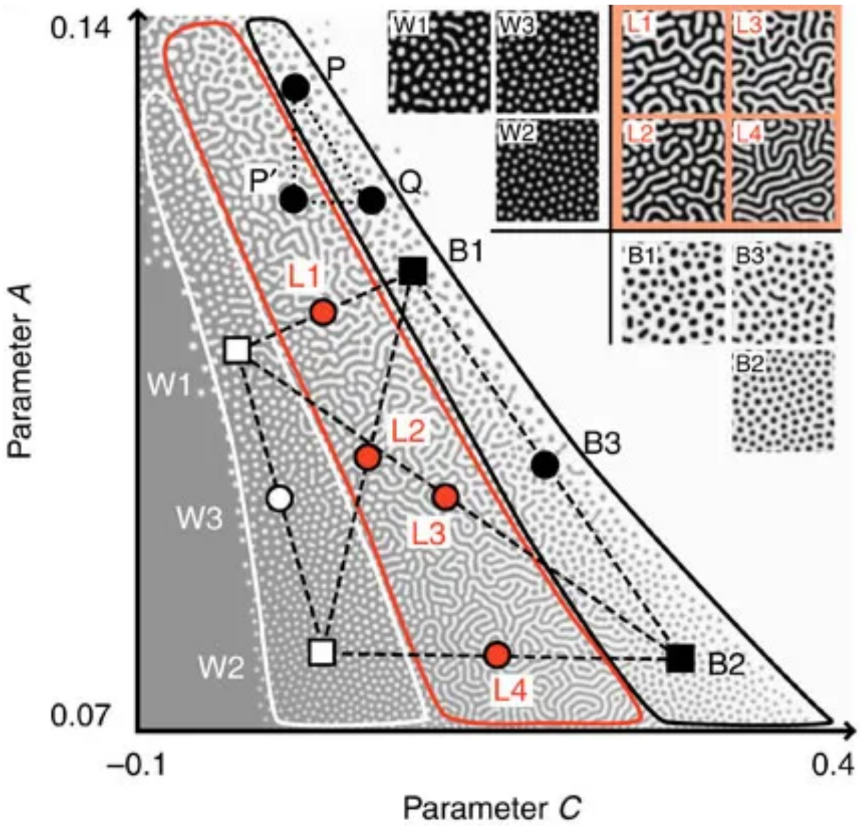


Figure 3.6: Different patterns obtained by Miyazawa et al. [2010] through modification of parameters $A$ (autocatalysis) and $C$ (basal synthesis) in the linear reaction-diffusion system implemented in this chapter. *Image reproduced from Miyazawa et al. [2010]. Article is licensed under CC BY 3.0*

### 3.3.2 Migration

In the second process—migration—, a shape is grown by robots at the edge of the swarm, and in areas of low concentration of molecule $U$, orbiting around the swarm to reposition themselves in areas of high concentration of molecule $U$, i.e. in the proximity of Turing spots. There are three motion-related states the robot can be in: *WAIT*, *EDGE FOLLOW* and *RECOVER*. The default state is *WAIT*, where robots are stationary. For a robot to move, the first condition that it must satisfy is to be at the edge of the swarm. Edge detection is based on a cohesion metric obtained by comparing the running average number of neighbours of the robot with the running average number of neighbours of its neighbours. The intuition is that robots inside the swarm would have a similar number of neighbours as their neighbours, whereas robots at the edge would be surrounded by less robots compared to their neighbours, as some of them would be inside the swarm, therefore having a higher number of neighbours. Using information from neighbours produces a more robust measure of cohesion [Nembrini and Winfield, 2012]. Running averages increase robustness in the presence of noise in communication with neighbours. They are calculated as:

$$new\ average = \alpha \times current\ measure + (1-\alpha) \times running\ average \tag{3.7}$$

Equation 3.7 is used to estimate the running average number of neighbours of a robot at every time step, as well as the running average number of neighbours of its neighbours. The parameter $\alpha$ governs the speed of convergence, set to $10^{-4}$ in our experiments (after checking on real Kilobots, this value makes them adapt to changes in neighbours in approximately 10 seconds). If the ratio between a robot's running average number of neighbours and the running average number of neighbours of its neighbours is less than a predefined threshold (in our case, 0.8), the robot is considered to be at the edge of the swarm. This value was chosen after visual experimentation in simulation, and it maximised the number of robots correctly detected at the edge and minimised the number of robots wrongly detected at it.

After a robot detects itself at the edge of the swarm, it is only allowed to migrate if its concentration of molecule $U$ is relatively low (less than 4 units), it is not surrounded by neighbours with high concentration of molecule $U$, and there are no other neighbours moving (required for edge-following, as it will be explained below). These conditions make robots in areas of low concentration move, while preserving robots in areas of high concentration. If the conditions are met, the robot starts moving around the swarm in a fixed direction (clockwise in our experiments), and the robot is said to be in the *EDGE FOLLOW* state. To do so, the robot follows the edge of the swarm by always orbiting around its nearest stationary neighbour, as described by Rubenstein et al. [2014b], i.e. *"by maintaining a fixed distance to the center of the closest stationary robot"*. Orbit behaviour is achieved by the robot through rotation. If the distance to its nearest stationary neighbour is larger than the orbiting threshold $d_{th}$ (in our experiments, $d_{th} = 45mm$ for robots to orbit as near to neighbours as possible to maximise communication without crashing), the robot rotates in the opposite direction as it was rotating before, until this distance becomes smaller than the threshold. When this happens, the robot switches the direction of rotation again, until the distance becomes larger than the threshold, and so on. This is visually represented in figure 3.7. It is important to highlight that the center of rotation of the robot is directly below each motor (not the center of the robot itself), hence producing the desired orbiting behaviour. In addition, it is crucial that neighbours (or at least the nearest one) are stationary for edge-following to properly operate.

Due to malfunctioning or motion noise, a migrating robot might separate from the swarm, hence increasing the likelihood of losing connectivity. This happens when the distance to its nearest neighbour is larger than 60mm. If so, the robot enters the *RECOVER* state, which implies rotating around itself until it detects at least one neighbour. The robot keeps rotating in one direction until the distance to its closest neighbour increases—the robot has been approaching its nearest neighbour, but now it is going further away from it. Then, it switches direction to keep approaching its nearest neighbour. Whenever such distance increases, the robot switches direction

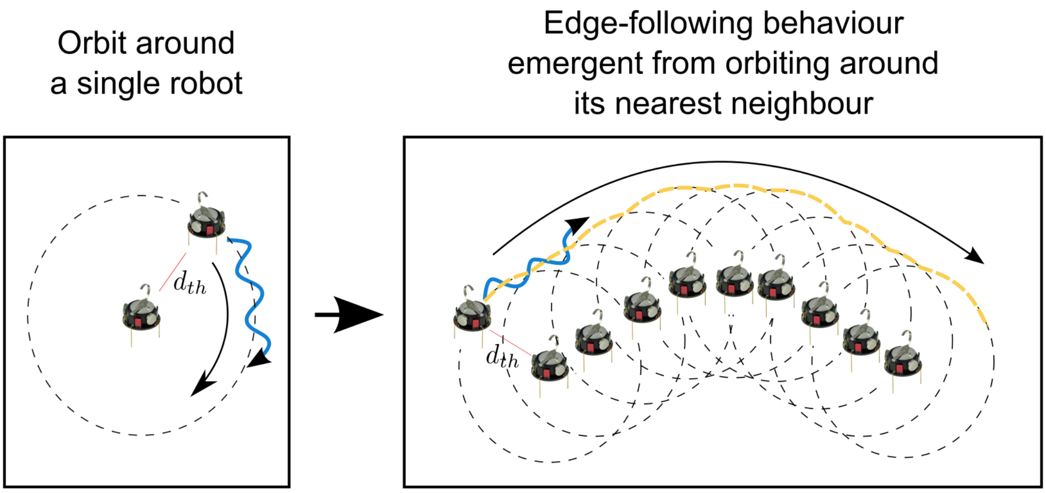


Figure 3.7: Edge-following behaviour emerges from simple orbiting behaviour around a robot's nearest stationary neighbour.

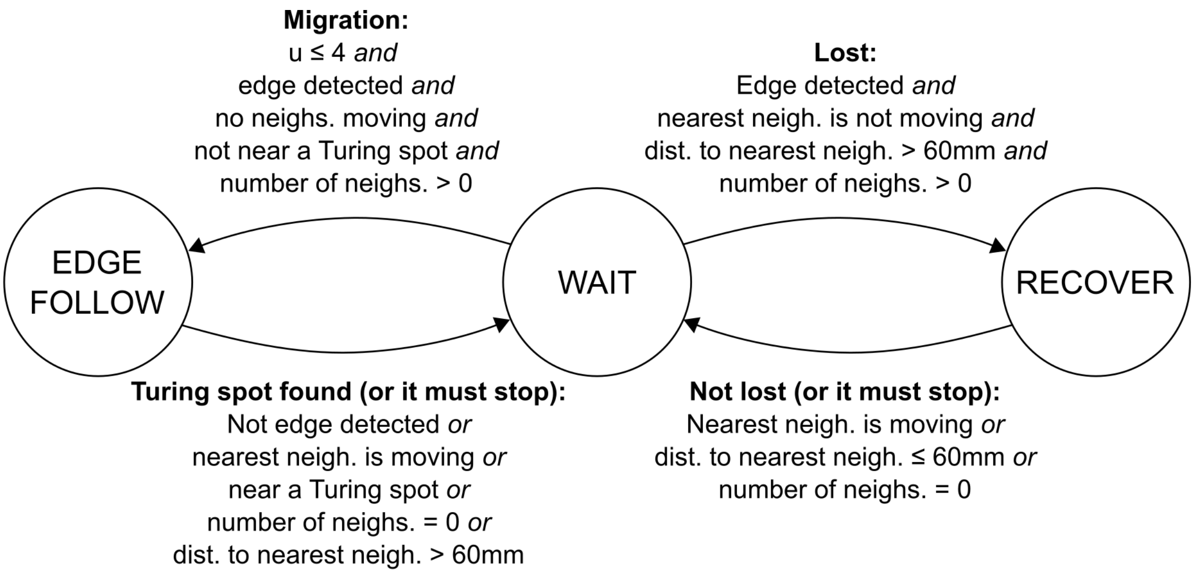


Figure 3.8: State machine of the three motion-related states robots can be in. In the *WAIT* state, the robot is stationary. It then transits to the *EDGE FOLLOW* state for migration (when the robot is at the edge of the swarm, and far from areas of high concentration of molecule $U$). If the robot gets lost, it enters the *RECOVER* state to reconnect with the swarm by rotating towards its nearest neighbour.

of rotation, until it reaches a safe distance of 60mm with respect to its nearest neighbour. Indeed, this is a similar behaviour as the self-assembly algorithm described by Gauci et al. [2014b].

Finally, when a migrating robot detects a Turing spot, e.g. a certain number of robots with high concentration of molecule $U$ (in our case, at least two neighbours), it stops next to them, switching to the *WAIT* state, as all stationary robots. This creates an accumulation of migrating robots around the Turing spot, hence growing protrusions from them. By running reaction-diffusion

**Algorithm 1:** Pseudo-code of bio-inspired morphogenesis

```
// Initialisation: random molecule concentrations, robot not moving, kiloticks = 0
setup()
// loop() function
while TRUE do
    /* It processes received messages, and updates neighbours' tables and running
       averages of neighbours and neighbours of neighbours                        */
    receiveInputs()
    /* Concentration of molecules U and V is updated based on the linear model for
       reaction-diffusion                                                          */
    reactionDiffusion()
    if kiloticks > WAITING_TIME then
        /* Robots on the edge orbit the swarm until they find areas of robots with high
           concentration of molecule U, or recover from getting lost              */
        stateMachine()
    end if
    /* It shows the corresponding LED colour depending on the concentration of molecule
       U                                                                           */
    showColour()
    // It computes a new, unique, local ID in case of clash
    checkLocalID()
    /* It removes the entries from the neighbours' table not updated in the last s
       seconds                                                                     */
    purgeNeighbours()
    /* It updates the message that it is sent with its state and molecules concentration
       */
    updateMessage()
end while
```

patterning alongside migration, migrating robots see their molecule concentration increased after they stop by the vicinity of Turing spots, eventually shifting the Turing spot towards the tip of the protrusion. This makes other migrating robots stop. Thus, it is the combination of patterning and migration that results in morphogenesis. A summary of all the states and their transitions is shown in figure 3.8. It is important to point out that the transitions are not fully symmetric for several reasons. For example, in the transition from *WAIT* to *EDGE FOLLOW*, the condition $u \leq 4$ only makes sense when the robot is in *WAIT* state, not when it is in *EDGE FOLLOW*, as robots do not update their molecules concentration when moving. Also, in the transition from *EDGE FOLLOW* to *WAIT*, the condition *dist. to nearest neigh.* $> 60mm$ only makes sense for that transition (and for *WAIT* to *RECOVER*) to prevent the robot from getting lost. Finally, a condition for robots to enter the recovery mode is to be at the edge (to avoid robots inside the swarm to wrongly enter this mode). However, when they are recovering from being lost, they are allowed to break this condition if this makes them return to the swarm.

**Algorithm 2:** Pseudo-code of the NSCP morphometric

**Input** : Array $P$ containing all points of the shape contour; Angles threshold $t$ between 0 and 180 degrees

**Output** : Array $Q$ containing points in the same order from $P$ with angles $< t$ for all three consecutive points

```
Q ← P
if Q contains at least 3 points then
    base ← 1
    for k ← 2 to length(Q) do
        // Q is circular. When k = length(Q), then Q_{k+1} = Q_1
        angle ← compute internal angle from points Q_base , Q_k and Q_{k+1}
        if angle ≥ t then
            Q ← remove point Q_k from Q
        else
            base ← k
        end if
    end for
end if
return Q
```

The corresponding pseudo-code of both patterning and migration processes described in these subsections is shown in algorithm 1.

### 3.3.3 Morphometrics selection

We decided to use morphometrics to quantitatively compare morphologies in order to better describe the effect of the morphogenesis algorithm here proposed on the shapes produced by the robot swarms. Several morphometrics have been proposed by biologists, as described in §2.1.4. Among them, we decided to use Shape Index and the Number of Shape Characterising Points due to their simplicity and suitability for shapes growing from a circle, as in the case of our robot swarms. Furthermore, both metrics are dimensionless, hence facilitating the comparison with the shapes created by swarms with a different number or type of robots.

For the Shape Index, only the perimeter and the area of the shape has to be calculated. All the points on the contour of each shape were obtained by using the *findContours* function included in OpenCV 3.2.0 using *CHAIN_APPROX_NONE*. Then, we used the functions *contourArea* and *arcLength* of OpenCV 3.2.0 to calculate the area and perimeter of the contour of the shape.

The NSCP metric was also chosen due to the fact that it can give a sense of the number of protrusions that arise in the shape. To calculate it, the length of the array resulting from applying algorithm 2 to the array of all contour points with a threshold of 160 degrees was obtained.

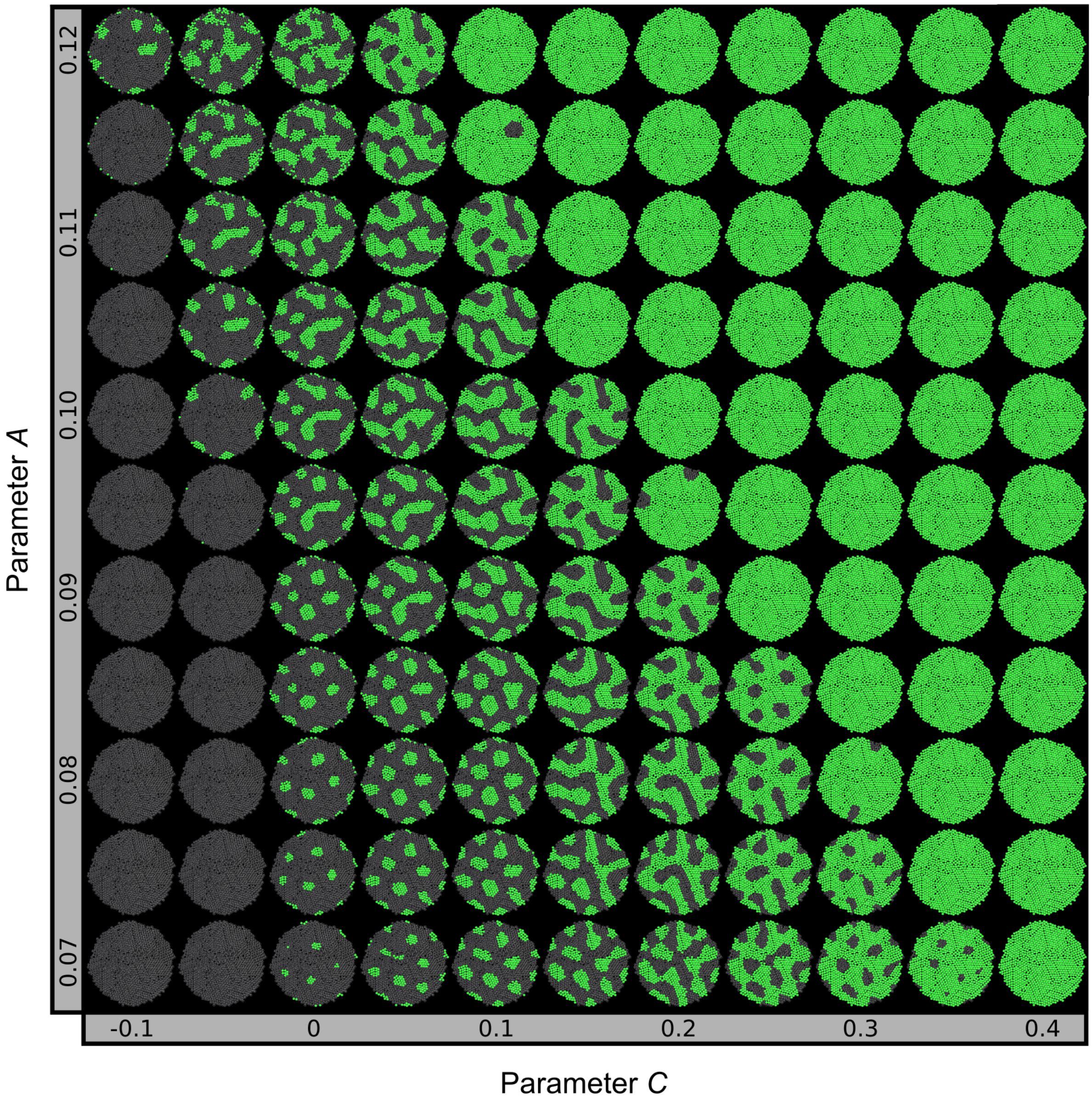


Figure 3.9: Stable patterns obtained with 121 static swarms of 1000 simulated Kilobots performing patterning (without the migration process) with the linear reaction-diffusion system. Green colour indicates that the concentration of molecule $U$ in the robot is higher than 4 units, whereas no colour (i.e. grey) means that concentration is less than or equal to 4 units.

## 3.4 Results

In this section, I present results in terms of emergence of shapes, adaptability to different initial configurations and robustness to different types of damage in experiments with a large number of Kilobots, as well as experiments to explore the different types of patterns that can be achieved in simulated swarms. Experiments with real robots took between 3 and 4 hours in total.

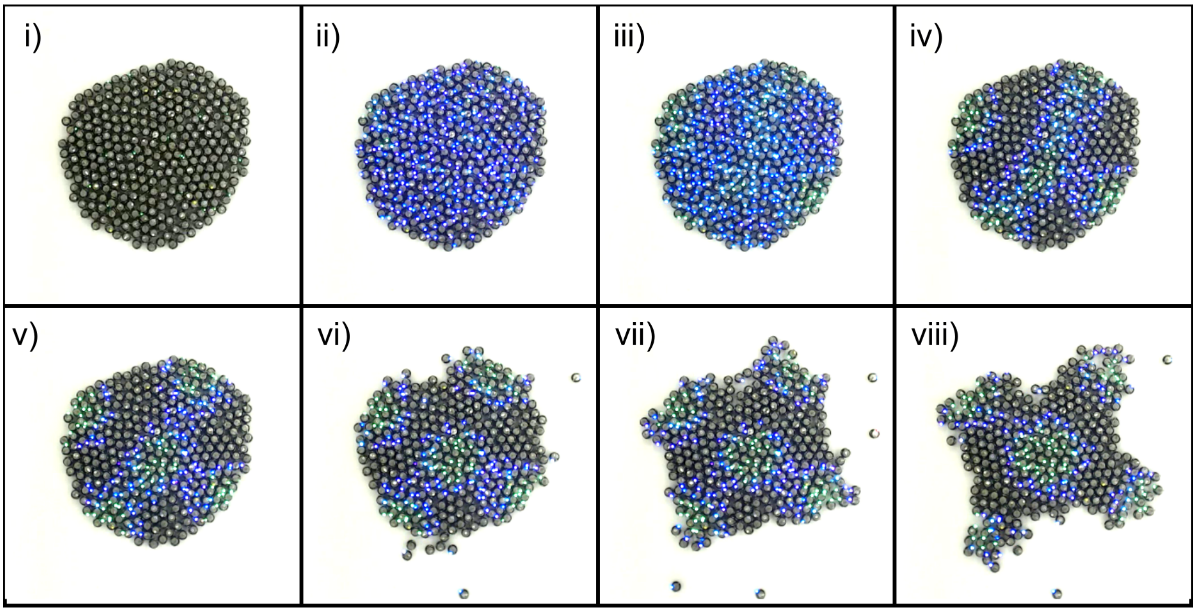


Figure 3.10: A sequence of a swarm of 300 Kilobots performing morphogenesis. The swarm was initially placed as a circle. Then, five Turing spots emerged from the patterning process, with four of them at the edge of the swarm and another one in the middle. Finally, robots grew four protrusions by extending the four Turing spots on the edge, resulting in a regular, cross-like morphology.

### 3.4.1 Exploration of parameters

We explored the same parameter space as the study by Miyazawa et al. [2010], i.e. parameters $A$ (autocatalysis of molecule $U$) and $C$ (basal synthesis of molecule $U$). The goal was to adjust the reaction parameter $R$ to replicate the same patterns obtained by the previous authors, hence successfully implementing the linear reaction-diffusion system in the swarm. As shown in figure 3.6, parameters $A$ and $C$ of the linear reaction-diffusion system control the type of patterns that emerge, from spots to stripes and inverted spots. A total of 121 simulations of swarms of 1000 Kilobots were performed. Parameter $A$ ranged from 0.07 to 0.12 (similar to Miyazawa *et al.*'s work, as they explored from 0.07 to 0.14), and it was tested every 0.005 units. We decided that the range 0.12 to 0.14 was not important to investigate because the region of patterns at this range is very narrow (see figure 3.6). Parameter $C$ ranged from -0.1 to 0.4 (directly taken from the work by Miyazawa *et al.*), and it was tested every 0.05 units. The rest of parameters were left with the same coefficients as described in §3.3.1. We found that a value of $R = 160$ achieved very similar patterns as the ones shown in Miyazawa et al. [2010]. Results are shown in figure 3.9. This confirmed that Turing patterning can be achieved in simulated swarms of robots using only local communication, as well as the different types of patterns that can be achieved with a simple linear system, e.g. spots, inverted spots, stripes, or a combination of them. We chose to use parameters producing spots to have a good compromise between the number of migrating

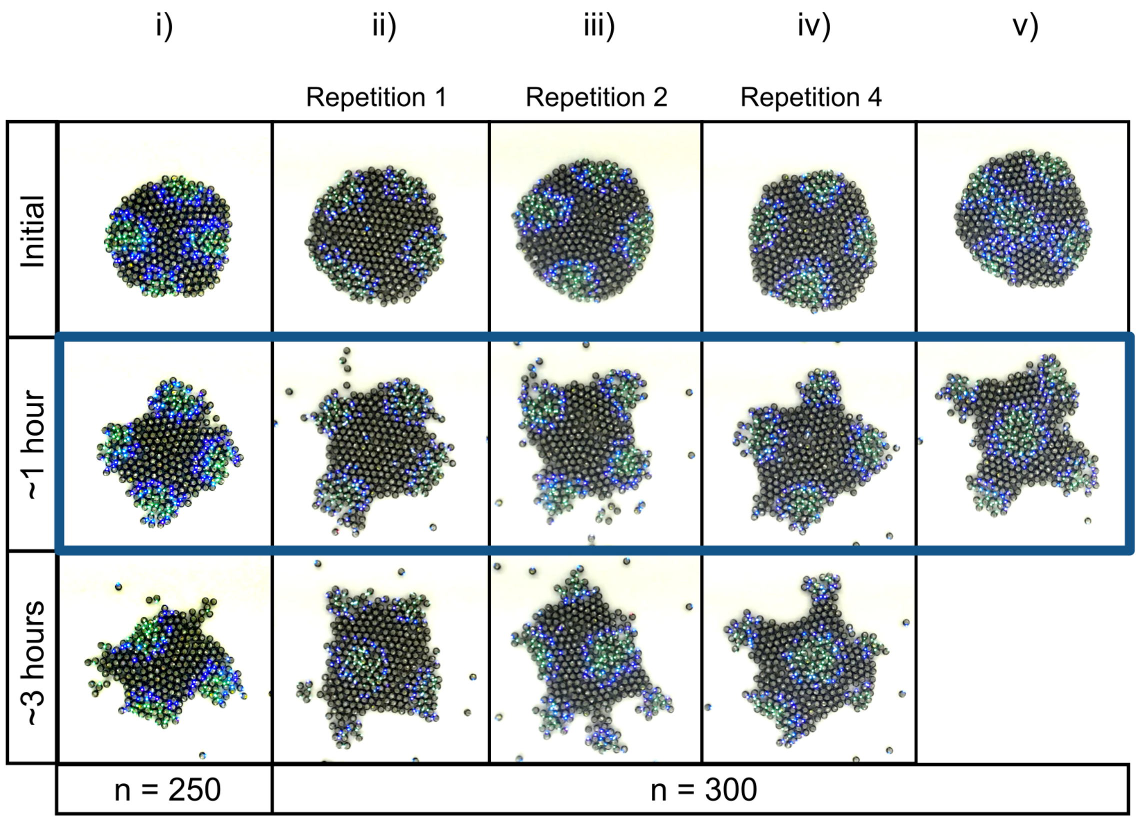


Figure 3.11: Summary of the five experiments with circular initialisation producing regular morphologies (highlighted in blue in the second row) during morphogenesis. Snapshots of the initial pattern before migration, and shapes after approximately 1 hour and 3 hours are shown, as well as the number of robots in each experiment. Experiments *i-iv)* lasted approximately 3 hours and a half, whereas experiment *v)* lasted about 1 hour and a half.

robots and the areas of growth. Sections below show that Turing patterning can also be achieved in real swarms of simple robots.

It is important to highlight that the parameters of the reaction-diffusion system define the separation between spots/stripes, their size, etc. Thus, the same style of pattern will be formed regardless of the number of robots in the swarm. The difference lies in how much of the pattern can be seen. For example, swarms of 300 robots will have less spots than swarms of 1000 robots. However, the size of those spots and separation between them will be the same.

### 3.4.2 Emergence of morphologies

We validated the morphogenesis approach on swarms of 300 Kilobots (although one experiment featured 250 robots), initially placed as a circle in the middle of the arena. We were interested in testing whether the morphogenesis algorithm could produce emergent swarm morphologies. By *emergent morphologies* here we only mean the ability of swarms to transform into other morphologies different from their initial configuration in a self-organised way by growing protrusions.

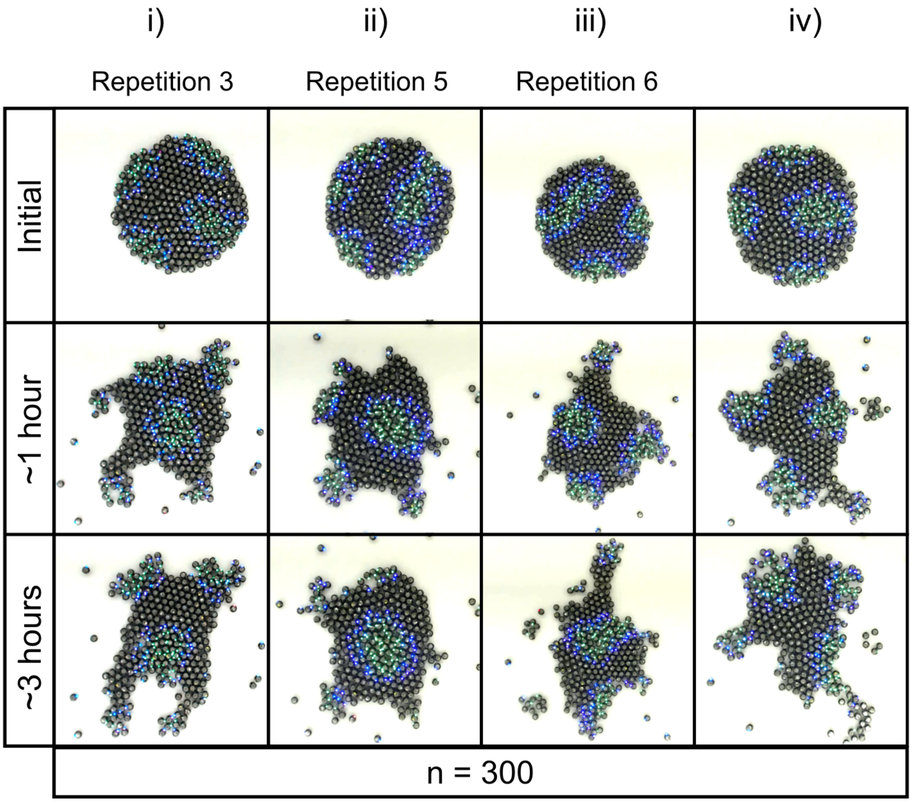


Figure 3.12: Summary of the four experiments with circular initialisation producing organic morphologies during morphogenesis. Snapshots of the initial pattern before migration, and shapes after approximately 1 hour and 3 hours are shown, as well as the number of robots in each experiment. Experiments lasted approximately 3 hours and a half.

A total of nine experiments with an initial circular shape were performed. Of these, three were performed to adjust the other parameters of the algorithm (e.g. number of neighbours with $u > 4$ that a robot must sense to detect itself near a Turing spot), and six were repetitions with the same parameters. Running time was approximately three hours and a half in average for all the experiments (except for one of the experiments adjusting parameters). In the temporal sequence from initialisation to morphogenesis of one of those experiments shown in figure 3.10, we can see that the emergent morphology is regular, i.e. the four protrusions grew in a similar way, as theorised (see figure 3.3). This means that we could predict how growth would occur given the initial spot-like pattern, i.e. protrusions grew exactly from the spots. Occasionally, some robots got lost from the swarm due to noise, but most of the robots remained within the swarm, hence not affecting morphogenesis. Five out of the nine experiments produced this regular shape half-way through the experiments, as seen in figure 3.11. Swarms also produced not-so-regular, organic-like shapes. By organic shapes we mean shapes that did not have regular protrusions growing from the initial spot-like pattern at any point during the experiment, but unpredicted growth instead. This is shown in figure 3.12. Therefore, shapes could not be as controlled as

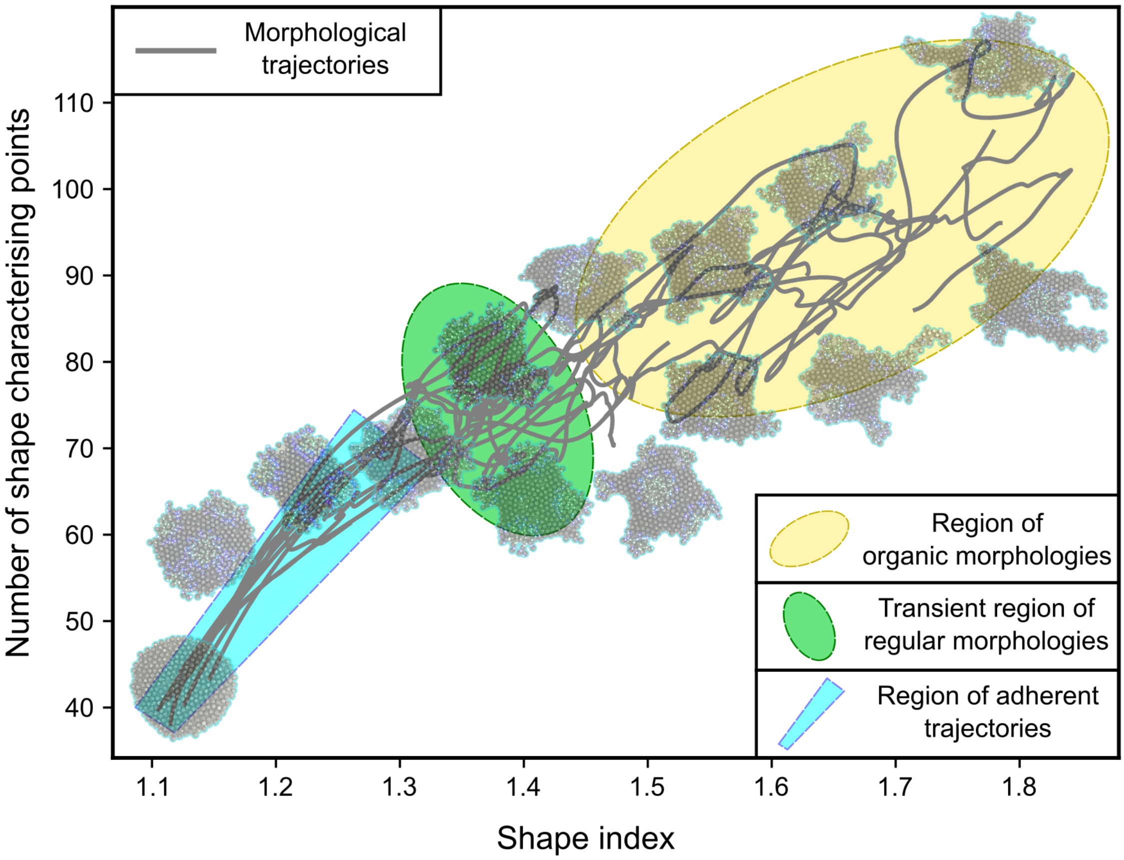


Figure 3.13: Morphospace of the nine circular experiments with respect to Shape Index and the Number of Shape Characterising Points. Trajectories of the evolution of shapes over time are shown as grey lines, as well as some shapes during morphogenesis. Three regions are identified: a region of adherent trajectories with similar growth, a transient region of regular morphologies, and a region of diversification with organic-like shapes.

expected. This had to do with instabilities, which I explain in more detail in §3.5.

We used the chosen morphometrics, i.e. Shape Index and the Number of Shape Characterising Points, to quantify the morphologies produced by the swarms in the nine experiments. Evolution of the shapes with respect to both morphometrics was tracked from the beginning to the end of the experiments. Three regions can be identified: i) a first region of adherent trajectories, where evolution is very similar across the nine swarms, ii) a transient region where regular morphologies are found, and iii) a region of organic morphologies where swarms developed a wider range of organic-like morphologies. The trajectory of the swarms through the morphospace over time is shown in figure 3.13. Evolution of the morphologies was individually quantified in terms of Shape Index, as shown in figure 3.14.

We also wanted to check whether the combination of patterning plus migration derived in morphogenesis, or instead migration alone would still produce morphogenesis. In other words,

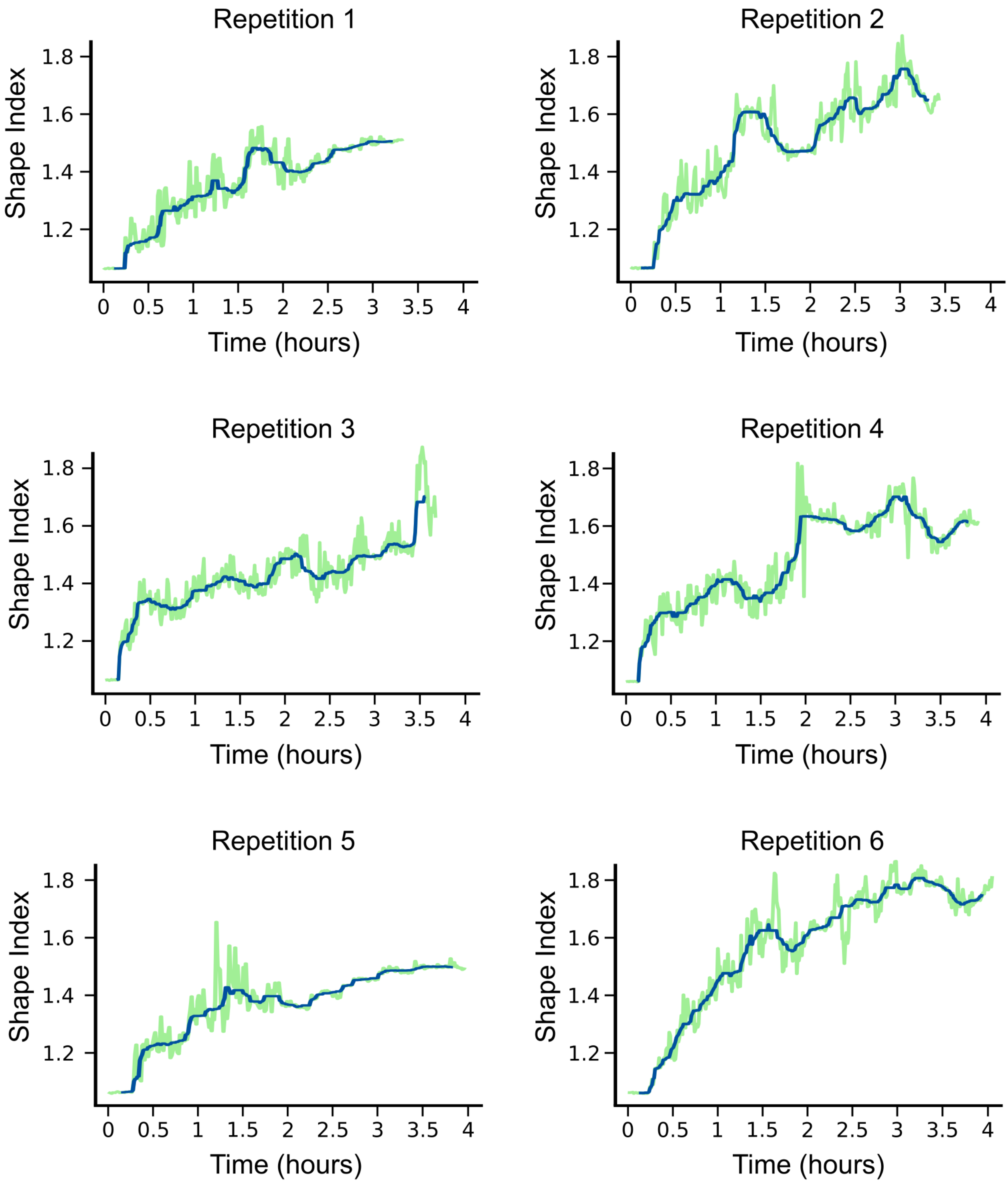


Figure 3.14: Graphs showing the Shape Index in green and median Shape Index of a moving window of 13 minutes (centered at the current time on the x axis) in blue of the six repetitions of the morphogenesis algorithm starting from a circular shape.

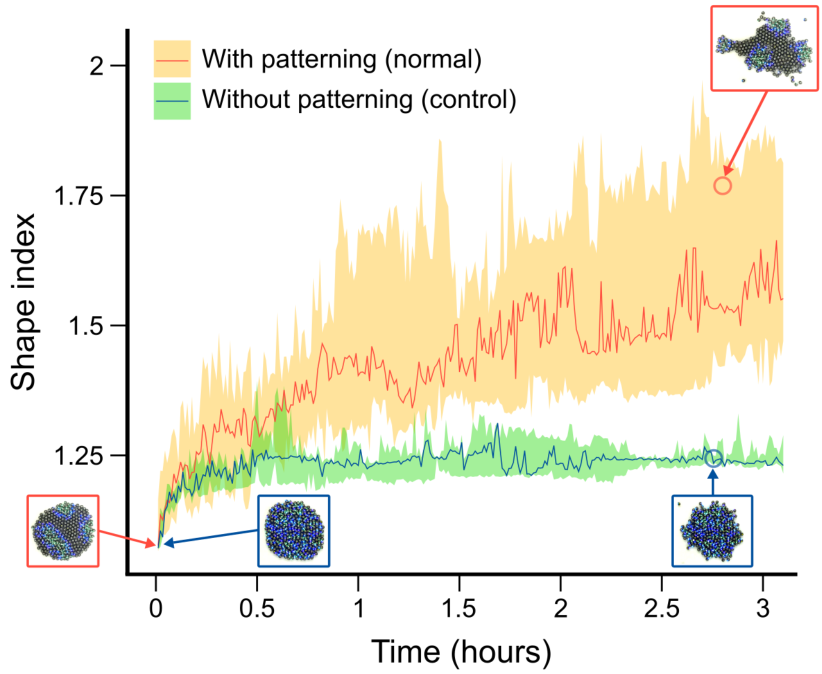


Figure 3.15: Comparison between morphogenesis with patterning (normal, orange ribbon at the top) and without patterning (control, green ribbon at the bottom). The graph shows the median, absolute maximum and absolute minimum Shape Index every forty seconds of the six repetitions of the Turing morphogenesis algorithm starting from a circular shape, and three repetitions of morphogenesis with random concentrations, i.e. migration with no previous Turing patterning.

we wanted to assess the role of patterning in the morphogenesis process. To test this, three extra experiments were performed with patterning switched off—robots only did migration with the same random concentration of molecules throughout the experiment. All parameters were kept the same as in the six repetitions. A quantitative comparison between these three extra experiments (the control experiments) with a random pattern and the six repetitions in terms of Shape Index was done. Swarms with a random pattern did not grow much after initialisation, whereas the swarms of the repetitions grew steadily throughout the experiments. In fact, the minimum Shape Index of the repetitions was already higher than the maximum Shape Index of the control experiments by the middle of the runtime. This shows that reaction-diffusion patterning is essential for the emergence of morphologies, and that shapes cannot grow from a random pattern. The corresponding graph is shown in figure 3.15.

### 3.4.3 Adaptability

In the swarm robotics community, adaptability (or flexibility) refers to the ability of the swarm to adapt to different environments and tasks [Brambilla et al., 2013]. However, in this thesis, by adaptability I mean the ability of the swarm to grow shapes even if started from different initial shapes. After the circular experiments were carried out, we were interested to see whether the

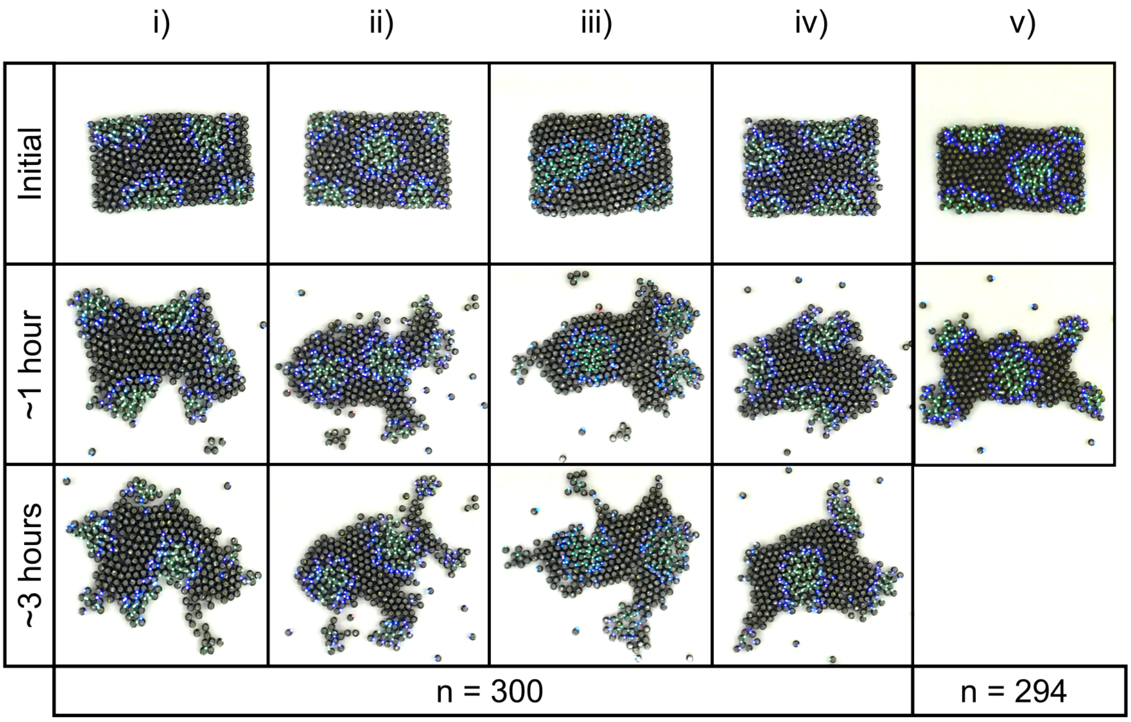


Figure 3.16: Summary of the five experiments with rectangular initialisation. Snapshots of the initial pattern before migration, and shapes after approximately 1 hour and 3 hours are shown, as well as the number of robots in each experiment. Experiments *i-iv)* lasted approximately 3 hours and a half, whereas experiment *v)* lasted about 2 hours.

swarm could still grow shapes if started from another initial configuration different from a circle. For that, we performed five experiments with a rectangular shape as the initial configuration of the swarms, and run the algorithm with the same parameters as used in the six circular repetitions. Figure 3.16 shows the initial, middle and final shapes of these experiments. Furthermore, to quantitatively test whether there was any difference between the circular and rectangular experiments, we compared them in the same way as in the comparison between the morphogenesis algorithm with and without patterning. At the beginning, the rectangular experiments start with a Shape Index equals to approximately 1.20, whereas the circular experiments have an initial Shape Index of approximately 1.05, as they are not a perfect circle. However, swarms from both sets soon grew indistinguishably, hence showing that the algorithm is adaptable in the sense that it can grow morphologies with different starting configurations. A graph showing the evolution of the Shape Index in the circular and rectangular experiments is shown in figure 3.17.

Another important question was whether a morphodynamic process could be observed in the experiments, i.e. the pattern adapting to the morphology and vice versa. We identified two examples of this phenomenon. The first example corresponds to the growth of a protrusion by maintaining its position at the tip of the growing protrusion at all times, even though such tip was composed of different robots over time. In the second example, a Turing spot moved from the

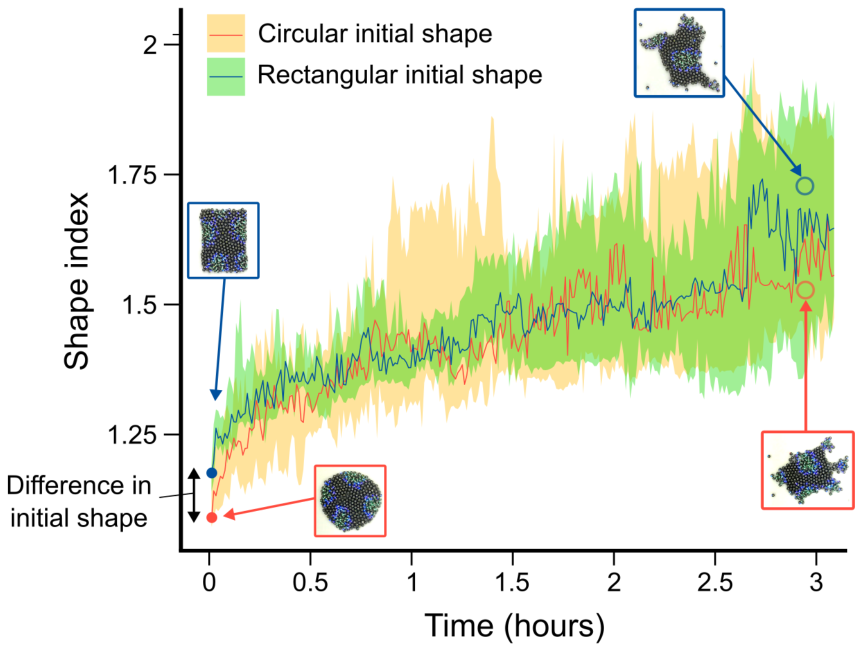


Figure 3.17: Comparison between the experiments starting with a circular configuration (orange ribbon at the bottom) and rectangular configuration (green ribbon at the top). The graph shows the median, absolute maximum and absolute minimum Shape Index every forty seconds of the six repetitions of the Turing morphogenesis algorithm starting from a circular shape, and five repetitions of the same morphogenesis algorithm starting from a rectangular shape.

right-hand side edge towards the middle of the swarm. Another Turing spot appeared near the previous location of the moving spot, hence producing a more stable configuration of the pattern for the reaction-diffusion system. Figure 3.18 shows the two examples of the pattern shifting during morphogenesis explained before. These examples show that the morphogenesis algorithm dynamically makes the swarms adapt during the process of morphogenesis itself.

### 3.4.4 Response to damage

Finally, we also showed the ability of the morphogenesis algorithm to recover from different levels of damage to the swarm. Although the term *robustness* lacks a formal definition by the swarm robotics community [Bjerknes and Winfield, 2013], it generally means the ability of the swarm to cope with all types of failures, from failures of the individual robots (e.g. hardware) to failures on task completion at the swarm level. In this thesis, by robustness I solely mean the ability of the swarm to self-repair when it is perturbed, i.e. regrow missing parts or continue growing even if there are missing robots. Two experiments were performed with low level of damage (cutting off protrusions), and one experiment with higher level of damage (cutting the swarm in half). In the first case, a growing protrusion is cut off while leaving some robots in the Turing spot. By the end of the experiment, the swarm had regrew the protrusion. In the second experiment, two

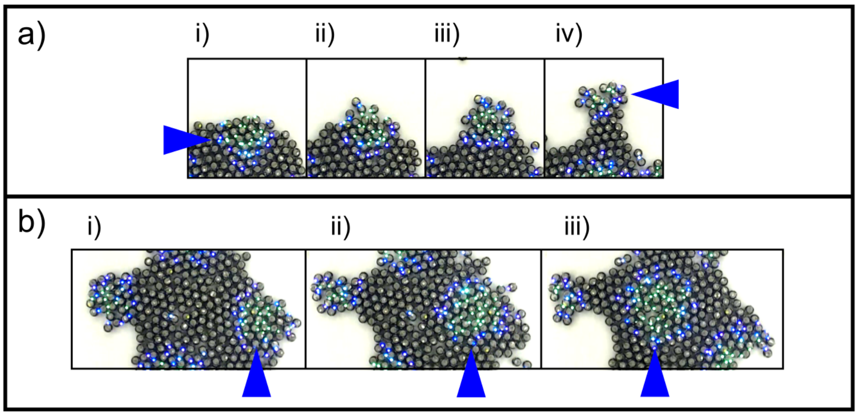


Figure 3.18: Examples of patterning adaptation in response to developing swarm morphologies. *a)* The Turing spot is maintained at the tip of the protrusion. *b)* A Turing spot moves from one edge to the middle of the swarm, while other appears where the former was originally located.

complete protrusions were removed from the swarm, completely removing the Turing spots. In this case, more robots were at the edge of the swarm, hence helping migration—and subsequent growth—towards the top of the swarm. In the final experiment, the whole swarm was split up in two parts (with a separation between the parts of approximately 3 robots). By the end, the swarm managed to reconnect through a Turing spot, restoring connectivity. Figure 3.19 shows the results of these experiments.

## 3.5 Discussion

The previous results were achieved with large swarms of real, simple robots, reacting locally to their neighbours. In fact, swarms were composed of unreliable robots with imperfect motion, communication noise, inaccuracy in the estimation of the distance, etc. Furthermore, swarms could still grow shapes even if an average of 10% of robots got lost during migration—this is a reminiscent of embryo development, where some cells die in the process. This required the swarm as a whole to be greater than the sum of its parts, hence, giving rise to the emergent behaviour of creating different morphologies. Indeed, swarms created the expected regular morphology over half of the times in the circular experiments (see figure 3.11). Other morphologies resembling some primitive organisms were also created by the swarms (see figure 3.12), showing a wide range of shapes that can be produced by the morphogenesis algorithm. The advantage is that no map of the shape is required for the swarms to create it, as opposed to other approaches in large robot swarms [Rubenstein et al., 2014b; Gauci et al., 2018]. Instead, shape growth is produced by the combination of patterning and migration, resulting in the feedback loop described as dynamic morphogenesis in the literature [Salazar-Ciudad et al., 2003; Salazar-Ciudad and Jernvall, 2004].

Dynamic morphogenesis has been shown to play a crucial role in the development of shapes (see figure 3.18). As in biology, this feature makes the swarm more dynamic and adaptable. Tip

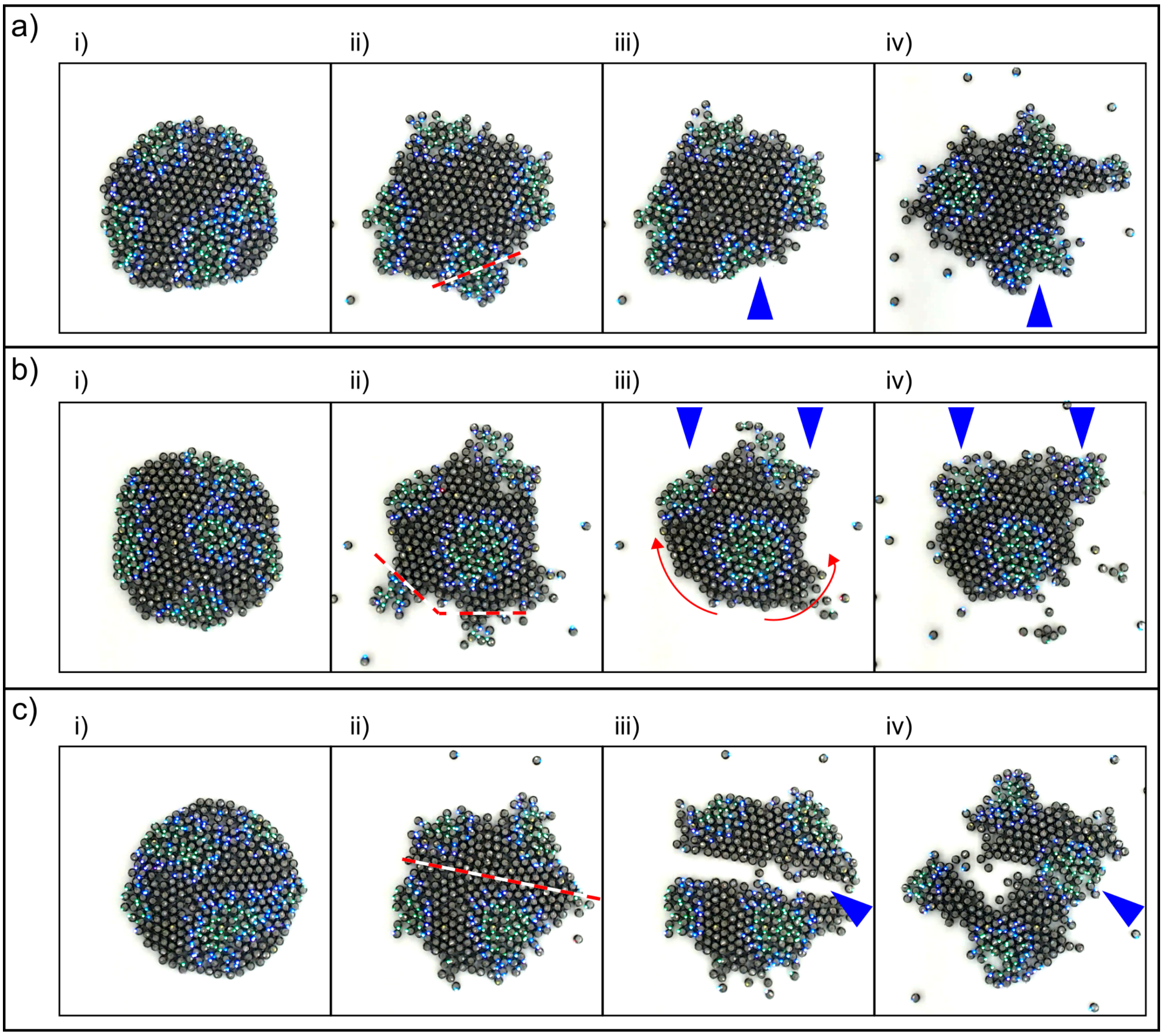


Figure 3.19: Robustness to different levels of damage. *a)* Minor damage to the swarm: part of a protrusion is removed. The swarm manages to regrow the missing protrusion. *b)* Minor but higher damage to the swarm: two complete protrusions are completely removed. Growth is produced at the top of the swarm, where a higher number of migrating robots can now reach. *c)* Major damage to the swarm: the swarm is split up in two. By the end of the experiment, the swarm reconnects through a Turing spot.

extension was the result of it. This meant migrating robots arriving to the vicinity of Turing stops, stopping by them, and seeing their concentration of molecule $U$ increased by the fact of being surrounded by neighbours with high concentration of molecule $U$. If no morphodynamic feedback loop had been in place (i.e. patterning switched off before migration), the pattern would have been static, hence causing the Turing spots to gradually become hidden by migrating robots. Eventually, migrating robots would continue orbiting the swarm indefinitely, as there would not be any Turing spot at the edge of the swarm. We also showed how important patterning itself is in the experiments with a random pattern (see figure 3.15). Therefore, it is the interplay between patterning and migration what resulted in self-organised morphogenesis.

A drawback of this morphodynamic process was found in the form of instabilities, i.e. when the pattern adapted to the growing morphology through an abrupt reorganisation of the Turing spots. During some of the experiments, we realised that instabilities in the pattern occurred while the shape was growing, hence disrupting growth due to the morphodynamic loop between patterning and migration. Half-way through the experiment, the pattern suddenly goes through an instability phase where the concentration of the activator molecule $U$ drops in all the swarm to very low levels (LED switched off, corresponding to concentrations of up to 1 unit). Then, it increases to very high concentrations (LED colours green and blue, corresponding to concentrations of between 3 and 6 units), to finally converge to a stable pattern of Turing-spots again—slightly different from the pattern before the instability. Sometimes, the observed behaviour is the opposite: the instability causes the swarm to have very high concentration, and then it drops, eventually creating Turing spots. Robot migration continues while the instability is taking place, as robots currently cannot detect when the instability is happening. Figure 3.20 visually shows an example of an instability.

Our hypothesis is that instabilities occur when Turing spots are too stretched out or too close from each other. The parameters of the reaction-diffusion system define how far apart the spots can be from one another. This means that the frequency of the Turing spots remains the same no matter the size of the surface. However, the morphodynamic loop makes the pattern adjust to the new morphologies, hence modifying the pattern. The pattern might not meet the conditions defined by the reaction-diffusion equations and their parameters any more, resulting in a sudden adjustment of the pattern by creating/joining spots.

Instabilities negatively affected controllability of the shapes. Swarms could not get the same level of controllability in the shapes as in Rubenstein et al. [2014b] or Gauci et al. [2018]. Part of this can be explained because the pattern cannot be fully specified—it is completely self-organised. The parameters of the reaction-diffusion system define the type of pattern that will emerge (i.e. spots, stripes or a combination of them), so one can control the number of spots/stripes, the separation between them, etc., but not the specific arrangement/location of them. However, disruptions in the pattern due to instabilities played a major role in the lack predictability and controllability of the shapes. On the other hand, instabilities could be advantageous for swarms

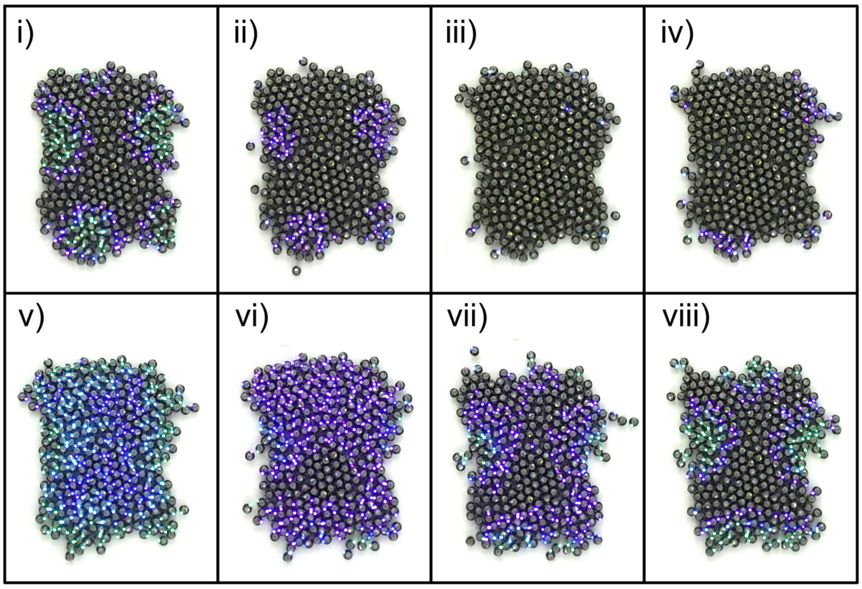


Figure 3.20: Time sequence of a swarm of robots going through an instability in the pattern during morphogenesis. *i)* Pattern before the instability. *ii-iii)* Concentration of molecule $U$ drops. *iv)* Concentration of molecule $U$ starts to increase. *v)* Molecule $U$ reaches high concentration across the swarm. *vi-vii)* Turing spots start to emerge again. *viii)* The instability is finished, and a slightly different pattern emerges in the swarm (a wide Turing spot at the bottom, as opposed to two Turing spots).

if they could be controlled as desired. For example, an instability could be triggered to disrupt a static pattern emerged from reaction-diffusion in order to dynamically reconfigure the swarm if more exploration is needed, or to create a new pattern if the swarm gets stuck by obstacles in the environment. Therefore, instabilities might be a powerful tool for adaptability. This remains as an open line of research for the future.

Finally, the morphogenesis algorithm has also been shown to be adaptable to two different initial configurations: a circle and a rectangle (see figure 3.17). In these experiments, rectangular swarms started from a Shape Index slightly higher than the circular swarms (because this metric measures how close a shape is to a circle). By the end of the running time, the shapes that were generated from both sets of experiments almost overlapped completely with respect to Shape Index, meaning that swarms could produce practically the same type of shapes regardless of their initial configuration. However, it is important to acknowledge that the initial difference between the circular and the rectangular experiments was barely 0.15 units in their Shape Index. In order to better assess the adaptability of the swarms to different initial configuration, it is important to study other initial shapes with a higher difference in Shape Index. Such study would conclude whether swarms can grow in the same fashion independently of their initial shape, or growth is limited by their initial configuration. This remains as future work.

Swarms were also able to continue growing even in the presence of minor and major damage (see figure 3.19), showing a high degree of robustness to perturbations, as also shown in other approaches in the literature [Mathews et al., 2017; Cheng et al., 2005; Thenius et al., 2011; Shen et al., 2004; Meng et al., 2013; Jin et al., 2012; Sayama, 2010]. Indeed, adaptability and robustness are very important features for swarms to be deployed in real-world applications.

## 3.6 Concluding remarks

In summary, we have endowed large swarms of real, noisy robots with a self-organised morphogenetic behaviour by drawing inspiration from principles of developmental biology. The morphogenesis algorithm proposed in this chapter has been shown to produce emergent shapes, be adaptable to different starting configurations and the morphodynamics of the pattern, and be robust to damage through self-repair. Instead of using explicit information such as the shape to create or preprogrammed robots, patterning based on a reaction-diffusion system was used to allow completely self-organised morphogenesis to emerge, as seen in natural morphogenesis [Raspopovic et al., 2014]. The process of patterning was combined with robot migration to emulate flow of cells [Keller, 2005], producing a morphodynamic process of pattern and morphologies adjusting to each other, as also seen in nature [Salazar-Ciudad et al., 2003; Salazar-Ciudad and Jernvall, 2004].

A total of 121 simulations with swarms of 1000 robots have been performed to find the parameters that replicated results from the linear reaction-diffusion system [Miyazawa et al., 2010] and choose the best ones for our purposes. Then, a total of 20 experiments with swarms of cca. 300 Kilobots were conducted to validate our approach with real robots. In particular, in this chapter I showed i) the wide range of regular and organic-like shapes that can be grown with the help of morphometrics, ii) how important patterning is for shape growth (robot migration without patterning does not produce shapes), iii) the fact that swarms grow in a similar fashion regardless of their starting configuration (either circular of rectangular), and iv) the ability of swarms to recover from damage in the form of cutting-off of protrusions or splitting the swarm in half.

This chapter represents a successful proof of concept of completely self-organised morphogenesis showing the dynamic and organic adaptability of living organisms, hence serving as the base for the development of a morphogenetic system. This could be highly beneficial for applications requiring adaptable and robust spatial behaviours, such as search and rescue, space exploration or biomedicine. In this chapter, shapes were not used with a particular goal in mind, but used to demonstrate the feasibility of the algorithm for real robot swarms. Although there is some degree of control over the Turing spots, shapes cannot be fully controlled. Controllability and functionality should be granted to the morphogenesis process first if spatially-organising robot swarms are to be deployed in real-world applications. This will be explored in the next chapter.

The following items should be considered for future work:

- It would be useful to understand how the instabilities are produced, whether there are other parameters spaces or topologies that are more robust to instabilities (e.g. Scholes et al. [2019]), and whether we could trigger them as desired.
- Other types of initial shapes and/or patterns should be explored to, for example, find the optimum ratio between robots in the Turing spots and robots able to migrate to increase performance/growth.
- Scenarios with obstacles should be tried to test the response of morphogenetic swarms to them.

CHAPTER 4

# Controllable morphogenesis in robot swarms

In the previous chapter, morphogenesis based on patterning and migration was shown to be a feasible strategy for completely self-organised shape formation in large and real swarms of simple robots. The morphogenesis algorithm was also shown to produce emergent shapes, to adapt to different initial configurations, and to recover from damage to the swarm. However, shapes could not be easily controlled as desired. For morphogenesis to be useful in future real-world applications, we should be able to control the shapes that emerge. In this chapter, a new, bottom-up morphogenesis algorithm to allow such controllability is presented. The main contribution is the local gradients algorithm during the migration process, which robots use to define controlled gradients. As in the previous chapter, the swarms self-organise into different shapes using only local interactions and without the need for any map, coordinate system, central control or seed robots, but in a more controllable fashion. Controllability is achieved through parametrisation, i.e. by introducing three parameters on the local gradients that directly impact shape growth.

This chapter begins by summarising relevant work that the controllable morphogenesis approach builds on, followed by sections describing the improved morphogenesis algorithm based on local gradients and results of over 2000 simulations and 5 experiments with swarms of 300 Kilobots. I then discuss results regarding emergence and performance of controllable shapes in terms of exploration of the environment, scalability when swarms of 100, 250 and 1000 are used, adaptability to static obstacles and robustness to damage (cutting off protrusions). Finally, I present preliminary results showing how to grant functionality to the algorithm, as well as possible future extensions to it.

The work presented in this chapter, and much of the written content (except functional morphogenesis from §4.5), has been published in the following peer-reviewed journal:

- Carrillo-Zapata, D., Sharpe, J., Winfield, A. F. T., Giuggioli, L., & Hauert, S. (2019). Toward controllable morphogenesis in large robot swarms. IEEE Robotics and Automation Letters, 4(4), pp. 3386–3393.

– **Personal contribution**: I devised the work with the supervision of S. Hauert, L. Giuggioli, A. Winfield and J. Sharpe. I performed all the experiments and analysed the data. Finally, I wrote the paper, incorporating my supervisors' feedback, as first author.

This work was also presented at the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2019). Most parts of sections §4.2, §4.3, §4.4 and §4.6, as well as some parts of §2.2 and §3.3.1, are reproduced verbatim from this publication.

## 4.1 Introduction

The field of morphogenetic engineering aims to engineer self-organised spatial organisation in systems through a guided, bottom-up approach. By providing these systems with controllability, they could be applied to solve real-world problems as an engineering solution, hence achieving functionality (as opposed to order without purpose [Doursat et al., 2013]). The aim in this chapter is to increase the controllability and functionality of the morphogenesis algorithm developed in chapter 3 (referred to as the *original morphogenesis algorithm*, for convenience, in this chapter).

As described in chapter 2, controllability in morphogenetic robot swarms (in the sense that the shape formation process leads to a desired shape), has been approached from different angles in the literature. For example, some authors have used morphology commands to define the way robot swarms grow shapes [Mathews et al., 2017; Liu and Winfield, 2014; O'Grady et al., 2012], or a map of the target shape either either pre-processed or explicitly given [Gauci et al., 2018; Werfel et al., 2014; Rubenstein et al., 2014b; Meng et al., 2013; Jin et al., 2012; Guo et al., 2011; Bai et al., 2008]. In a real-world scenario where the swarm needs to adapt to an unknown environment, having a predefined shape might not be good enough to meet the requirements of the application due to a decrease in adaptability to changing environments or robustness to perturbations. To overcome this, Mamei et al. [2004] used morphogen gradients based on hop-count between the source of the gradient and its neighbours. In their work, robots had a different interpretation to the morphogen signals, hence taking a positional information approach (as described in §2.1.2.1). By doing so, robots with minimal capabilities could achieve different shapes such as circles, rings and polygons in simulation. Agents did not need any predefined shape, and used only local communication to transmit the type of gradient signal and its hop count—not even distance or angle to neighbours. As authors say in their paper, morphogen gradients can be used to create *"nearly circular regions of controlled size"* if a robot acts as the source of such signal, which is propagated up to a certain number of hops.

Reaction-diffusion alone was not able to produce controllable growth in chapter 3 mainly due to lack of control on the development of the pattern. As spots in the original morphogenesis algorithm were the main drivers for shape growth, the idea of using positional information could be beneficial if used to control how such spots grow by means of morphogen gradients. Hence, positional information could be used in conjunction with the morphogenesis algorithm based on reaction-diffusion and migration described in the previous chapter for the purpose of controlling spots, while keeping self-organisation. J. Green and J. Sharpe's paper on how both reaction-diffusion and positional information could work together [Green and Sharpe, 2015] is the source of inspiration for adding controllability to the morphogenesis algorithm in this chapter, concretely mode 1.

As described in §2.1.2.3, combination mode 1 depicts the scenario where reaction-diffusion establishes a prepattern that is then interpreted differently according to the areas of high and low concentration of morphogens, i.e. interpretation based on positional information. This concept is used as the basis for a new completely self-organised morphogenesis algorithm that builds on the original morphogenesis algorithm. However, in this new version, patterning and migration are decoupled (i.e. not running alongside each other anymore). By doing so, reaction-diffusion is used to establish an initial molecular pattern (as in the original algorithm) that defines two different roles for robots depending on whether they belong to areas of high concentration or not, hence having a different interpretation to the pattern (as mode 1 in [Green and Sharpe, 2015] describes). Robots in high areas of molecular concentration send a morphogen signal to create local gradients, that are controlled through three parameters. Robots in low areas of molecular concentration migrate towards the other areas in a similar way as in the original morphogenesis algorithm. Therefore, by using positional information here I refer to the use of the mechanism of having different responses to a pattern. I do not imply that swarms use any map or coordinate system to achieve controllable morphogenesis. Indeed, results of experiments and simulations show how the self-organised, controllable morphogenesis algorithm gives rise to a wider range of morphologies than the original morphogenesis algorithm, and that it can overcome obstacles and recover from damage.

## 4.2 Methodology

The algorithm is composed of two main processes: patterning and migration, to achieve morphogenesis. During patterning, clusters of attracting robots emerge from random initial conditions by a mechanism of reaction and diffusion of virtual molecules, as in the previous chapter. The patterning process then stops, hence defining clusters of robots maintained through local gradient signals. The emergent clusters are elongated by other robots stopping around them performing migration. This provides the swarm with the ability to grow a shape in a structured but completely self-organised fashion. A general overview of the algorithm is given in figure 4.1.

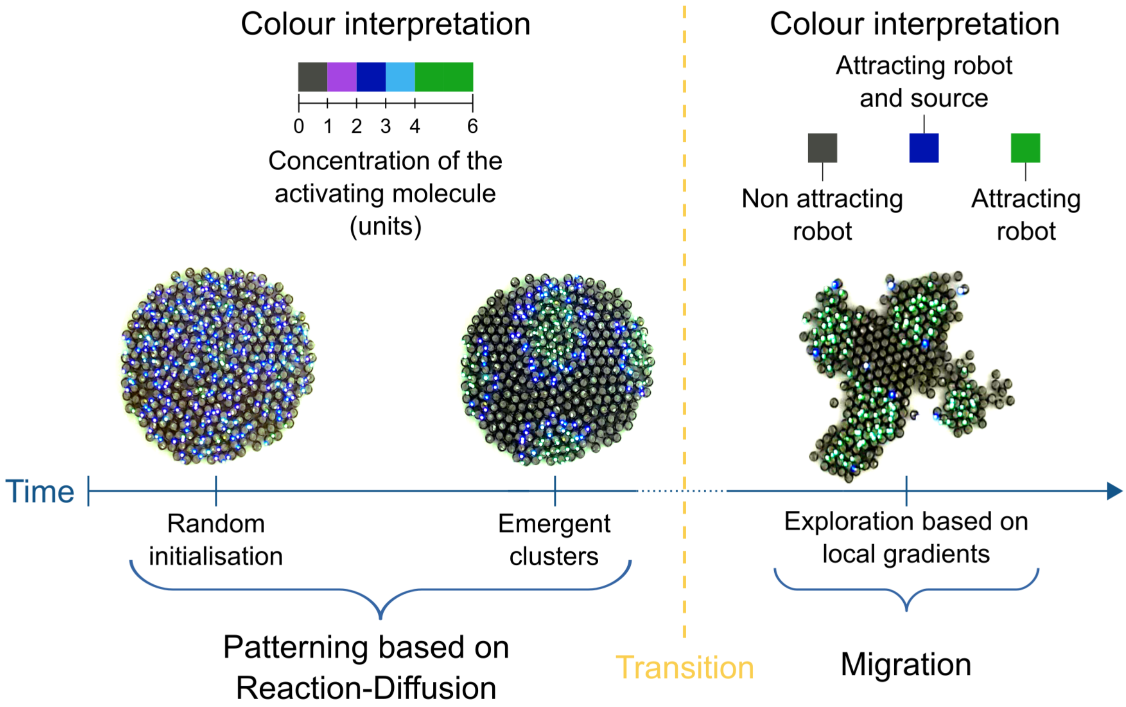


Figure 4.1: Overview of the controllable morphogenesis algorithm. Two-process schematic where migration occurs after patterning. Color interpretation is different for the two processes.

Before I dive into implementation details, it is important to clarify the terminology I use in this chapter. By *cluster of attracting robots* I mean a group of neighbouring robots where moving robots will stop nearby. *Attracting robot* and *attractor* refer to the same type of robots, those being stationary (not moving) and part of a cluster. In these clusters there will always be a robot sending a hop-count signal during the migration phase, known as the *source*. A detailed description of each process (patterning and local gradients) is given below.

### 4.2.1 Pattern formation

For this process, the same linear approximation of a reaction-diffusion system is used with the same parameters and structure described in the previous chapter (see §3.3.1). It is worth mentioning that another patterning strategy could potentially be used (for example, small clusters made of source robots spread out by a minimum distance from each other). Here I chose reaction-diffusion patterning due to its self-organised nature, and its ability to create spots on the edge of the swarm with the coefficients I use. In the unlikely event of no spots on the edge, robots could rerun patterning as many times as desired. This behaviour has not been implemented here, but left as a future extension.

All robots execute the patterning process during approximately 8 minutes in real time (60000 kiloticks in the simulator Kilombo, controlled with variable *PATTERNING_TIME*) to leave the reaction-diffusion system enough time to develop. Even though 32 kiloticks equal to 1 second in the real Kilobots, I measured that approximately 125 kiloticks in simulation correspond to one second in real time by comparing the time it takes the pattern to complete in both simulation and real robots. Finally, robots check whether the stable is stable in their own neighbourhood. This is explained in the next subsection.

### 4.2.2 Self-organised transition

Robots automatically detect whether the pattern is locally stable by calculating the cumulative change in their own concentration of molecule $U$ during windows of two minutes. If the cumulative change is below a threshold (1 unit), they transition. Figure 4.2 shows a boxplot with the cumulative change in concentration of molecule $U$ of 30 robots picked up at random from 30 different simulations. From this figure we can observe that a cumulative change below 1 unit corresponds to the swarm being substantially stable. On the contrary, a higher change corresponds to the early stages of pattern development. It is worth highlighting that using only one of the two molecules is enough, as they are both linked through the reaction-diffusion system, i.e. if concentration of molecule $U$ changes, concentration of molecule $V$ would change, and vice versa. In the same way, if concentration of molecule $U$ does not change, concentration of molecule $V$ would not change.

In this self-organised transition, the pattern can be seen as a form of automatic task allocation. If the concentration of the activator molecule $U$ inside a robot is above a certain threshold (3 units), the robot is considered to be an attractor in the next process. A threshold above 3 units, which corresponds to green and dark blue robots during patterning (see figure 4.1), guarantees that clusters are originally constituted by enough number of robots while allowing enough space between clusters for other robots to move. As a result of this process, clusters of attracting robots appear on the edge of the swarm (due to the specific set of parameters of the reaction-diffusion system implemented on the robots, as seen in the previous chapter). Robots with a lower concentration than the threshold are considered non attractors. Hence, clusters are made of attracting robots only.

### 4.2.3 Migration

In this process, reaction-diffusion stops and robot movement starts. For a shape formation behaviour to occur in the robots, they have to move towards other areas in the environment while maintaining connectivity at all times. Branching-out behaviour happens with the help of robots being attracted to the clusters of attracting robots that have arisen from patterning. Those clusters act as virtual docking points for other robots. When a moving robot senses enough adjacent attracting neighbours, it stops and may become part of the attracting cluster to continue attracting other robots (more details given in §4.2.3.1). Connectivity in the swarm is guaranteed by edge-following movement, described in §3.3.2. Edge following allows the robots to move around

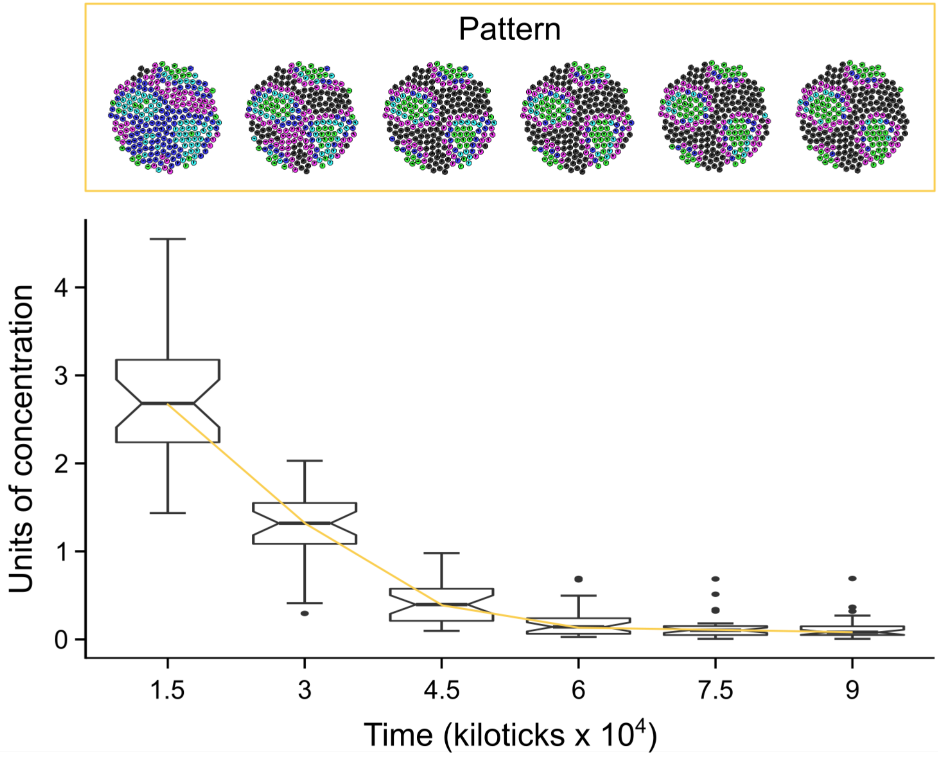


Figure 4.2: Cumulative change in concentration of molecule $U$ during patterning in windows of 15000 kiloticks in simulation (approximately 2 minutes in real time) with 30 robots at random from 30 different simulations with different seeds. Pictures of the same simulated swarm of robot are also shown above their corresponding time during the simulation.

the swarm while maintaining a fixed distance to the nearest neighbour (in this case, 45mm). Moreover, only the robots on the edge of the swarm which do not belong to any cluster are allowed to move. Whereas direction of movement was always clockwise in the original morphogenesis algorithm, in this chapter the direction of movement, i.e. clockwise or anti-clockwise, is randomly chosen with uniform probability $p = 0.5$ when the robot is initialised, and is fixed throughout the entire algorithm. This allows equal growth (probabilistically) on both sides of the branches.

Growth is achieved by having the attracting clusters at the tip of the branch at all times, as well as maintaining the size of the cluster within certain limits. If the clusters were too large or grew in size, other robots would be attracted to the base of the branch, hence, not producing the desired elongating behaviour. Robots fulfil those two conditions by creating a local gradient in the cluster. The last robot which arrives to the cluster (at the edge of the swarm) becomes the source of a signal which is transmitted inside the cluster of attracting robots to establish a hop-based gradient from itself. Robots calculate the lowest number of communication hops to reach the source of the gradient either directly or indirectly through their neighbours. As mentioned in §4.2.2, robots in the initial clusters are automatically considered to be attracting robots. Then, a non attracting robot becomes attracting robot but not source of the signal if it is closer to a source of a cluster than the threshold *MAX_HOPS*, and it has been an attracting robot at least once. This helps keeping the shape of the initial clusters for at least some time. If any robot became attracting robot by receiving the signal, then the edge of the swarm would soon saturate with attracting robots. Therefore, the only way for non attracting robots to become attractors is to orbit the swarm and find a cluster of attracting robots to become the source of the signal.

The message that a robot sends is composed of the robot ID (2 bytes), its number of neighbours (1 byte), the lowest number of hops to the source of the gradient, if in a cluster (1 byte), the ID of the robot that is the source of the local gradients, if in a cluster (2 bytes), the state of the robot (1 byte), and some binary information (1 byte) such as whether the robot is in a cluster, whether it is the source of a local gradients signal, and whether it is on the edge. A neighbours' table is used to store the information from a maximum of 20 neighbours (a good compromise between memory used and performance). The information stored is the random, unique, local ID from each neighbour, its distance, state, number of neighbours, whether it is an attracting robot, whether it is the source of a signal, the ID of the source of the signal in its cluster (if in any), the lowest number of hops to the source of the signal (infinite if not in a cluster), a timestamp of when the last message from this neighbour was received (entries older than two seconds are deleted to always keep up-to-date information), and its molecules concentration (used during the patterning process). Each robot has a 2-byte random, unique, local ID to make the algorithm completely scalable with respect to number of robots. If two neighbours share the same ID by chance at some point, they will generate random IDs until none of them share the same one—therefore, unique in their neighbourhood.

By combining a maximum number of hops to the source of the gradient with a mechanism whereby the last robot arriving to the cluster becomes the only source of the gradient signal, the robots self-organise to maintain the cluster at the tip of the branch (see figure 4.3). These clusters are indeed the nearly circular regions of controlled size pointed out by Mamei et al. [2004]. The robots with a number of hops to the nearest source higher than the threshold do not continue with the hop count, i.e. the signal is only transmitted inside the cluster to create a local gradient. Since such signal is not transmitted to the whole swarm, the process is completely scalable with respect to the number of robots in the swarm. Depending on the dynamics, any robot could potentially carry out any of the roles. Furthermore, no map, coordinate system or seed robots are required.

#### 4.2.3.1 Controllable morphogenesis

The goal in this chapter is to control the morphogenesis process while allowing self-organisation. This is achieved by having a range of variables that control the shape by modifying the local gradients. Five variables were initially proposed, and they are explained as follows:

- *Maximum number of hops to the source of the local gradient (MAX_HOPS).* It defines a hop-based threshold from the source of the gradient signal for stationary robots to belong

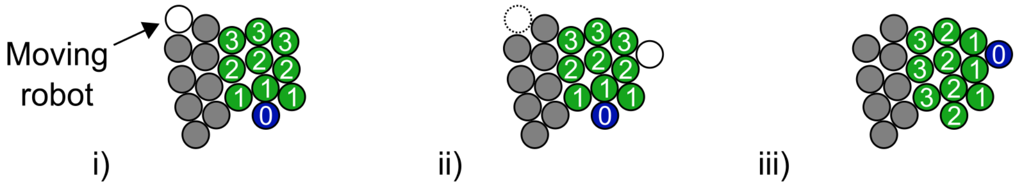


Figure 4.3: Diagram of a swarm with a cluster of attracting robots (in dark blue and green) and non attracting robots (in white and gray). A moving robot (*i)*) stops next to the cluster (*ii)*) and replaces the source of the signal (*iii)*). Hops are then readjusted to the new source. The hop-based threshold for belonging to the cluster results in the cluster remaining at the tip.

to a cluster of attracting robots. Therefore, robots must reach the source of the signal in *MAX_HOPS* hops at most to meet this condition (as well as the *MIN_FOR_ATTRACTOR* condition explained below). This variable is directly linked to the maximum size of the cluster.

- *Timesteps that the robot belongs to the cluster when it should not (COUNTER_NON_ATTRACTOR).* Each robot has a counter to avoid leaving a cluster straight away in case the source of the signal is suddenly lost, or the robot surpasses the *MAX_HOPS* threshold (when a new source arrives to the cluster but it is further away from the robot, for example). This variable enhances stability of clusters due to small fluctuations.

- *Minimum number of attracting neighbours to stop (MIN_TO_STOP).* It defines the minimum number of stationary neighbours inside an attracting cluster which a moving robot must sense to stop next to the cluster. This variable affects the location of the cluster where the robot stops, e.g. more on the sides or in the middle.

- *Minimum number of attracting neighbours to become an attractor (MIN_FOR_ATTRACTOR).* It is the minimum number of attracting neighbours which a robot (either during patterning or migration) must sense to become part of the attracting cluster (if it also meets the *MAX_HOPS* condition). This variable controls the minimum size of a cluster.

- *Max distance to attracting neighbours to become an attractor (MAX_DISTANCE_FOR_ATTRACTOR).* It is the maximum distance to attracting robots that a robot is allowed to be at to become part of the attracting cluster. This variable is linked to the previous one, as the *MIN_FOR_ATTRACTOR* attracting neighbours must be below the maximum distance specified by *MAX_DISTANCE_FOR_ATTRACTOR*. This variable controls how physically spread the robots in a cluster are allowed to be.

It is worth highlighting that *MIN_TO_STOP* and *MIN_FOR_ATTRACTOR* are independent of each other. A robot has only to satisfy *MIN_TO_STOP* for it to stop. When it satisfies *MIN_FOR_ATTRACTOR* it joins the cluster to be the source of the signal after it has stopped.

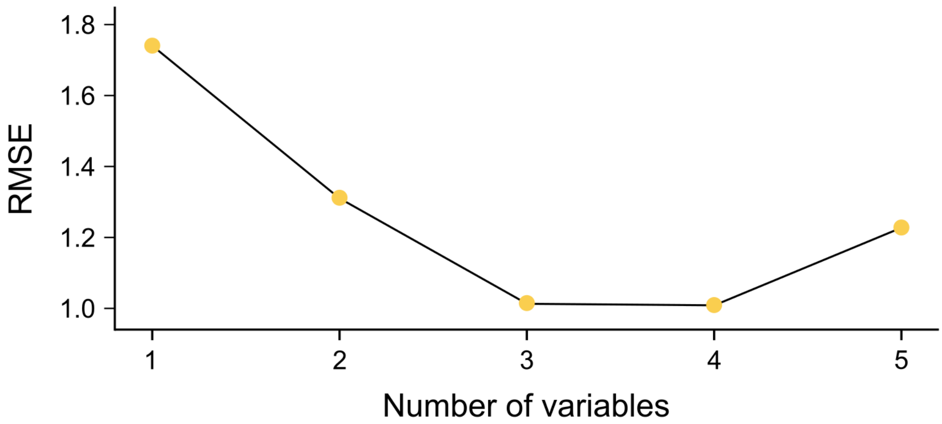


Figure 4.4: Random forests results for feature selection over the five different morphogenesis variables. The graph shows the Root Mean Squared Error (RMSE) of the total area covered during 20 repetitions per each combination of values of the variables. The number of variables refers to how many variables were used for prediction by incrementally adding one by one based on their ranking.

In case there are not enough attracting neighbours after stopping, the robot would still be non attracting. On the contrary, if the robot stopped next to a cluster for any other reason (e.g. when it senses another orbiting robot), it may become part of the cluster if there are enough attracting robots (at least *MIN_FOR_ATTRACTOR*) and it meets the *MAX_HOPS* condition.

A regression model based on random forests [Breiman, 2001] was used to select the most relevant variables, a common process in machine learning to reduce dimensionality known as *feature selection*. The performance measure was the total area covered during 20 repetitions of each combination of variables with different random seeds, as explained in §4.3. The random forest algorithm ranks the variables for their importance based on their predictive power, and then calculates the Root Mean Squared Error (RMSE) in the prediction of the area covered by adding the ranked variables one by one, starting from the highest ranked variable. The ranking of variables, from highest to lowest rank, was the following: 1) *MIN_TO_STOP*, 2) *MIN_FOR_ATTRACTOR*, 3) *MAX_HOPS*, 4) *COUNTER_NON_ATTRACTOR*, and 5) *MAX_DISTANCE_FOR_ATTRACTOR*. In the graph of RSME results of figure 4.4, we can see how the best number of variables to use is three or four variables, with barely any difference. I chose to use the three highest ranked variables (*MIN_TO_STOP*, *MIN_FOR_ATTRACTOR*, and *MAX_HOPS*) as the best variables for controllability. The fourth variable was not used to keep as few variables as possible, and also due to the fact that this variable would not help significantly. As a result, the variables *COUNTER_NON_ATTRACTOR*, and *MAX_DISTANCE_FOR_ATTRACTOR* were fixed for all the experiments. The counter was set to the equivalent of about 20 seconds in real time, and the maximum distance was set to 85mm (the communication radius).

The variable *MAX_HOPS* controls the maximum size of the clusters, *MIN_TO_STOP* controls

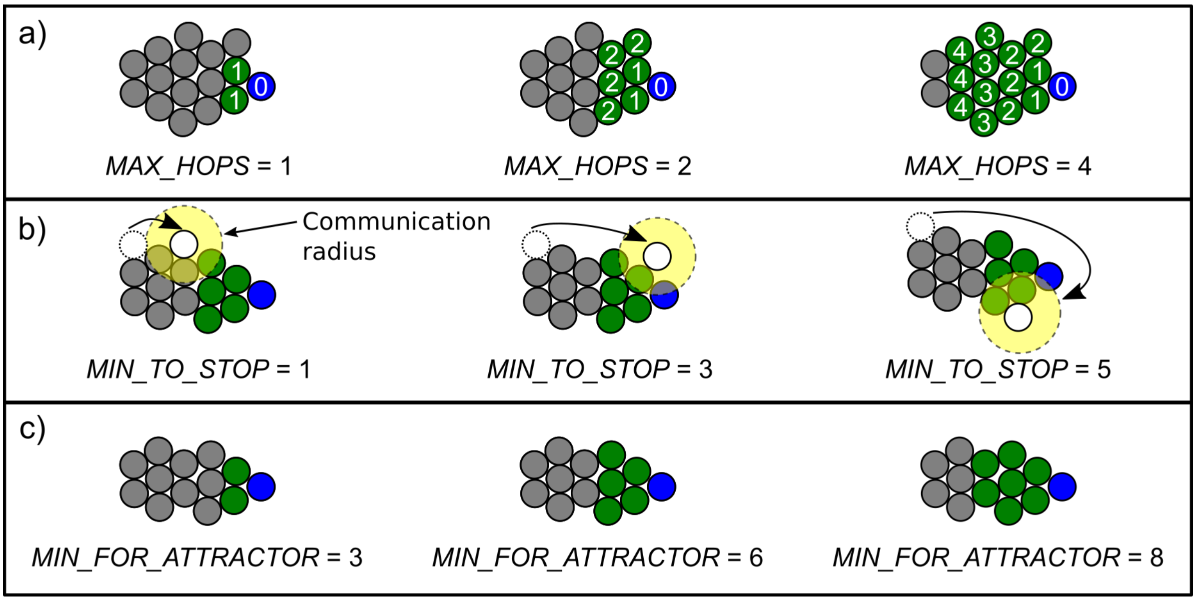


Figure 4.5: Effect of the chosen morphogenesis variables on the clusters.

where robots stop, and *MIN_FOR_ATTRACTOR* controls the minimum size of clusters. This is reflected in figure 4.5, which shows a diagram of the effect of the morphogenesis variables on the clusters of attracting robots. With the inclusion of these morphogenesis variables in the algorithm, the state machine from the previous chapter has been extended. Concretely, three new states have been added: non attracting robot, attracting robot but not source, and attracting robot and source. These new states extend the *WAIT* state from the previous chapter (see figure 3.8), as stationary robots can now be attracting robots (i.e. part of a cluster where migrating robots stop). Non attracting robots can either be stationary and not part of a cluster, or moving robots that are orbiting or recovering from being lost. Although a robot might be non attracting, if it has previously been part of a cluster and a new source arrives to the cluster, making the non attracting robot fulfil the hops condition, it will automatically become attracting robot, i.e. part of the cluster again (but not source). On the other hand, if a robot is non attracting but it fulfils the conditions to stop near a cluster after migration, it will become part of it by means of becoming the new source of the hop-count signal. Figure 4.6 shows a diagram with the state machine of the three new states for local gradients using the chosen morphogenesis variables. It is worth mentioning that there are other parameters and transitions not shown in the state machine for simplification. These add robustness to a system of unreliable, noisy robots. They are described as follows:

- Other robots on the edge of an attracting cluster might become sources of the local gradient signal with a certain uniform probability in case the source of the signal disappears. In the experiments, probability was set to $p = \frac{1}{8192}$, drawn at every timestep. This made the robots replace their sources in less than 10 seconds in real time, on average.

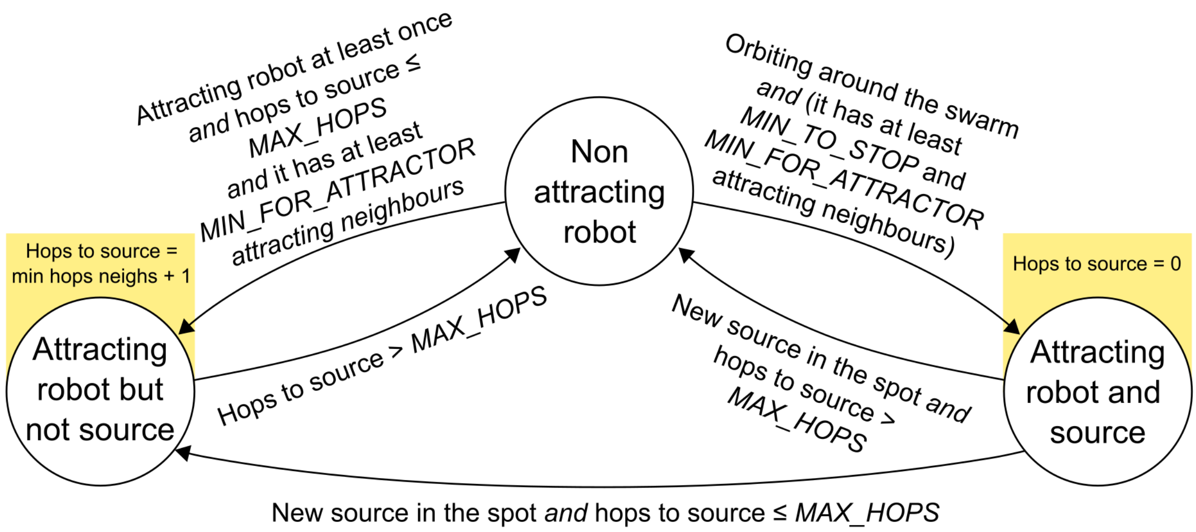


Figure 4.6: Finite state machine for local gradients. These states extend the *WAIT* state from the previous chapter (see figure 3.8).

- Apart from meeting the edge criterion, robots must have at least a certain number of neighbours at the edge in order to be considered to be at the edge. This helps with robots wrongly detecting they are at the edge inside the swarm. In the experiments, they must have at least one neighbour measuring itself as at the edge.

- Robots must orbit the swarm for at least a certain length of time to be allowed to become sources. This helps with robots suddenly waking up to orbit for a short period of time. In the experiments, this parameter was set to 8 seconds in real time. This value was chosen after manual experimentation with the Kilobots. Some of them wrongly tried to orbit the swarm when they should not, but realised of their mistake in less than 8 seconds.

- Finally, robots are considered faulty (hence, not being allowed to become sources ever) if they orbit around the same neighbours a certain number of consecutive times. For this, robots make a copy of their neighbours table at the beginning of their orbit motion. When they stop, they check how many different IDs they are surrounded by, compared to when they started moving. In the experiments, if they have less than three different neighbours for two consecutive times, they are considered to be faulty. This also helps with robots orbiting inside the swarm. This might be a problem if the IDs that are used are small. However, 2-byte IDs turned out to work well.

The pseudo-code of the extended morphogenesis algorithm with local gradients is shown in algorithm 3.

**Algorithm 3:** Pseudo-code of controllable morphogenesis

```
// Initialisation: random molecule concentrations, robot not moving
setup()
// loop() function
while TRUE do
    /* It processes received messages, and updates neighbours' tables and running
       averages of neighbours and neighbours of neighbours */
    receiveInputs()
    // PATTERNING
    if kiloticks < PATTERNING_TIME then
        /* Concentration of molecules U and V is updated based on the linear model for
           reaction-diffusion */
        reactionDiffusion()
        patternIsStable ← FALSE
    // SELF-ORGANISED TRANSITION
    else if not patternIsStable then
        /* It works out the cumulative change in concentration of molecule U during a
           window of w minutes */
        sum_u ← cumulativeChange()
        if sum_u ≤ 1 then
            patternIsStable ← TRUE
            if u ≥ 3 then
                state ← ATTRACTING_ROBOT
            else
                state ← NON_ATTRACTING_ROBOT
            end if
        end if
    // MIGRATION
    else
        /* Non attracting robots on the edge orbit the swarm until they find clusters of
           attracting robots. The latter maintain the clusters through local gradients.
           Robots also try to recover if they get lost. */
        localGradientsMigration()
    end if
    // It shows the corresponding LED colour depending on the process
    showColour()
    // It computes a new, unique, local ID in case of clash
    checkLocalID()
    /* It removes entries from the neighbours' table older than s seconds */
    purgeNeighbours()
    /* It updates the message that it is sent depending on the process */
    updateMessage()
end while
```

Table 4.1: Tested values of the variables for controlling morphogenesis and the best values found.

| Variable name | Values tested | Best values |
|---|---|---|
| *MAX_HOPS* | 0, 1, 2, 3, 4 | 0, 1 |
| *MIN_TO_STOP* | 1, 3, 5, 7, 9 | 5, 7, 9 |
| *MIN_FOR_ATTRACTOR* | 1, 2, 3, 4 | 1, 2, 3 |

## 4.3 Results

While a full analysis of reaction-diffusion patterning and adaptability to growth was explored in the previous chapter (see §3.4), here I focus on the controllability and performance aspects of the morphogenesis algorithm proposed in this chapter.

### 4.3.1 Emergence of controllable morphologies

#### 4.3.1.1 Simulated swarms

A parameter sweep across all combinations of plausible values was performed using simulator Kilombo to understand the effect of the morphogenesis variables quantitatively. Swarms of 250 simulated Kilobots were placed together in circular initial shape at the centre of a squared window of 850x850 pixels, initially covering about 8% of the total area. The choice of this number of simulated agents was made as a trade-off between the number of repetitions of each combination and running time of each simulation. In a similar way as in the work by Nembrini and Winfield [2012], area coverage was used as the output metric of the morphogenetic swarms. Performance of the different combinations was measured by calculating the total area of the window which the robots covered in an interval equivalent to 8 hours in real time (the amount of new area covered was continually stored at every timestep of the simulations). The purpose of the metric here was to measure how shape growth could be controlled in terms of by how much they could grow driven by the different combinations of variables. This controllability will be necessary to ultimately use morphogenetic engineering for specific applications in the future.

To decrease the reality gap between simulated and real Kilobots experiments, noise was added to both communication and motion. Each message had a 0.7 probability of being received correctly and not being discarded. Communication noise directly proportional to the distance between two robots was added with a Gaussian distribution of mean zero and standard deviation defined by the following equation:

$$\sigma_{distance} = \frac{d - min_d}{max_d - min_d} \tag{4.1}$$

In equation 4.1, $d$ is the distance measured by the robot, whose minimum is $min_d$ = 34mm (the diameter of the Kilobots) and maximum is $max_d$ = 85mm (the maximum communication

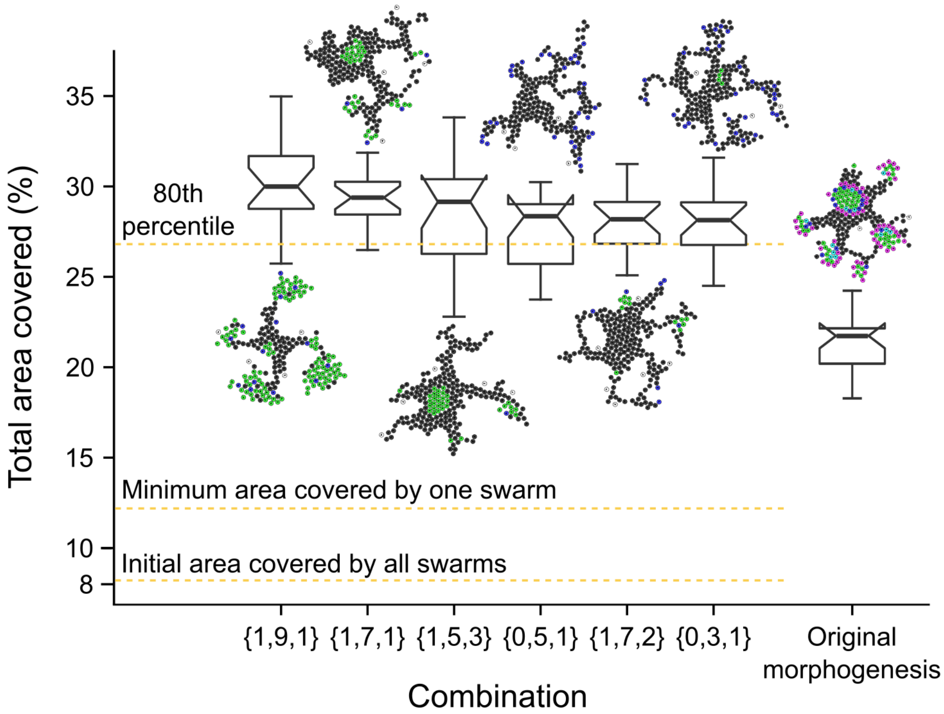


Figure 4.7: Box plot showing the best combinations of variables and the performance of the original morphogenesis algorithm from the previous chapter. The final shape of the best run for every combination is shown as well. Combinations are sets of {*MAX_HOPS, MIN_TO_STOP, MIN_FOR_ATTRACTOR*}. Dotted lines represent the corresponding area covered as described above them.

range allowed). As far as motion is concerned, every robot was initialised with a fixed bias in its angular velocity, randomly chosen from a Gaussian distribution of mean zero and standard deviation $\sigma_{\omega bias} = 0.01$. Moreover, another source of noise was added in the form of a Gaussian distribution of mean zero and standard deviation $\sigma_{\omega noise} = 0.005$ every time the simulator updated the movement of the robots. Noise values were chosen based on experimentation with real Kilobots. Robots which separated from the swarm due to noise were not taken into account for area coverage. Robots must have at least four neighbours to take part in the metric. The rationale behind is that lost robots would be unsuccessful in maintaining communication with the rest of the swarm, therefore, not contributing to a common shape.

A preliminary exploration was performed to identify the meaningful ranges for the morphogenesis variables. Table 4.1 shows the different values tested. From five *MAX_HOPS* upwards, swarms reached a static configuration due to most of the robots on the edge becoming part of a cluster soon after the start of the migration process. Regarding *MIN_TO_STOP*, robots on the edge usually sense an average of nine neighbours because density decreases compared to being in the middle of the swarm. Finally, the parameters of the reaction-diffusion system sometimes generate clusters of four attracting robots after patterning. If *MIN_FOR_ATTRACTOR* was higher,

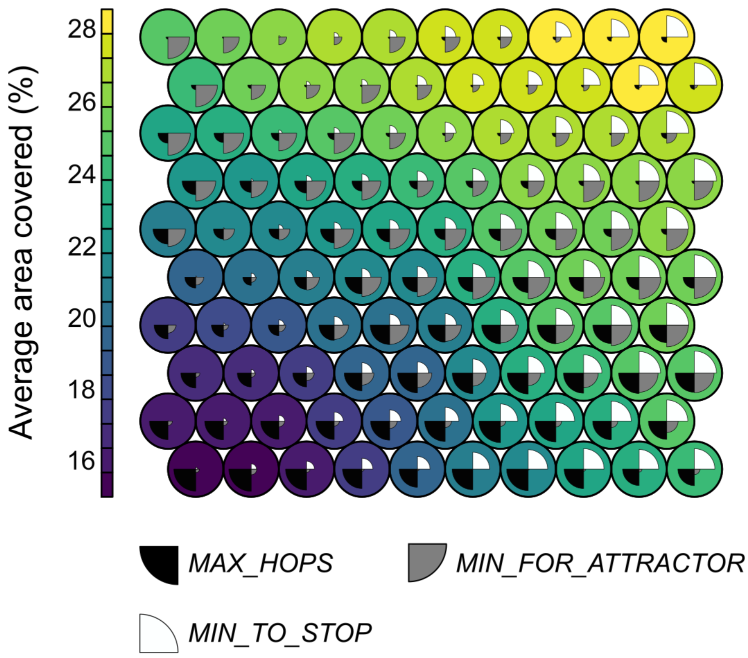


Figure 4.8: Self-organizing map with all combinations of the variables and a heat map of average performance across repetitions. Bigger wedges correspond to bigger values.

those clusters could not be formed, hence, losing attracting robots and chances for substantial growth.

For every combination, 20 repetitions with different seeds were performed. The same set of unique seeds was used for all combinations. Therefore, patterns emerged from reaction-diffusion were the same for all combinations (but not repetitions). This allowed to properly study the effect of the variables on morphogenesis under the same conditions. A total of 2000 simulations was performed. The 95% confidence interval for the median of the total area covered by the simulated swarms was calculated for each combination. The combinations with the highest, overlapping intervals of the median were identified. As a result, 6 out of 100 combinations were found to produce the highest performance in terms of total area covered (figure 4.7). A one-way analysis of variance (ANOVA) was done over the six combinations to test whether they were statistically different. This test rejected the null hypothesis ($F = 4.6281$, p-value $< 0.0007$), meaning that there were some differences in the means between those six combinations (with a 95% confidence level). A post-hoc Tukey test showed that the following combinations were statistically different at a 95% confidence level: {1,9,1} and {0,3,1} (p-value $< 0.015$), {1,9,1} and {1,7,2} (p-value $< 0.023$), {1,9,1} and {0,5,1} (p-value $< 0.0016$), {1,7,1} and {0,5,1} (p-value $< 0.045$).

Substantial growth did occur in simulated swarms. In the six best combinations of morphogenesis variables, swarms covered between three and four times their size while maintaining connectivity at all times, even if an average of 10% of robots got lost or became detached from the swarm by the end of the simulations. An extra 20 repetitions with the same set of seeds used for

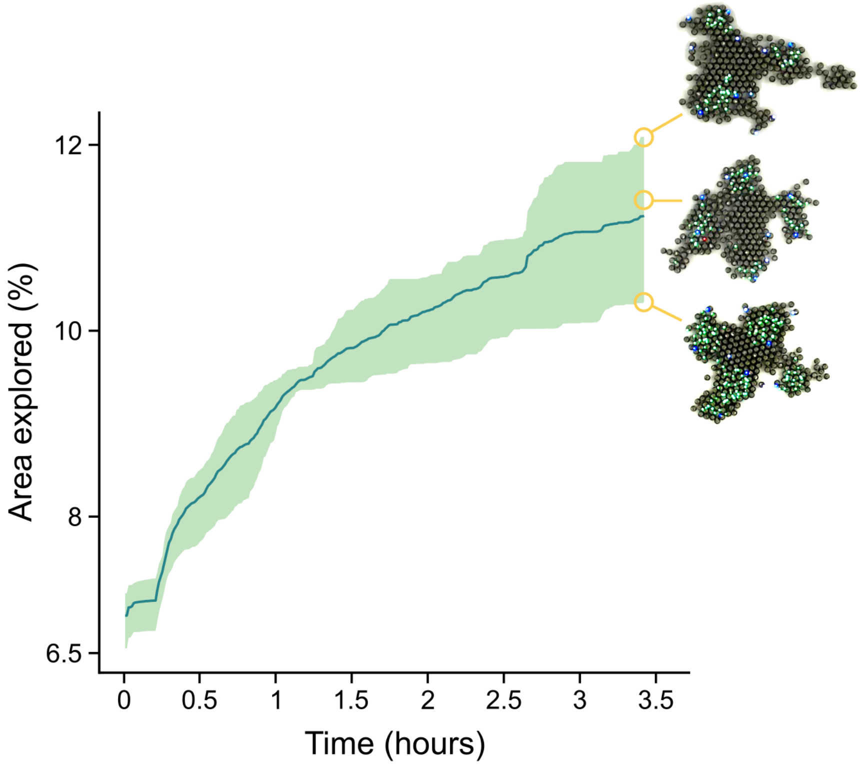


Figure 4.9: Percentage of area covered by swarms of 300 Kilobots. Minimum and maximum area covered by these swarms is shown as a green ribbon. Blue line in the middle of the ribbon is the average of three experiments with the same parameters.

each combination were done to compare the morphogenesis algorithm presented here with the one described in the previous chapter (also shown in figure 4.7).

Even though there is no statistical difference among the mean of some of the best combinations, a difference between them and the rest can be observed. Support comes from the fact that more than 50% of the repetitions (the median) of each combination covered a bigger area than 80% of all combinations and repetitions (the 80th percentile). Figure 4.7 shows this different effect of the morphogenesis variables depending on how they are combined. A self-organising map with all combinations was produced to further explore the effect of such combination of variables. As a conclusion of it, low *MAX_HOPS*, medium/high *MIN_TO_STOP* and low/medium *MIN_FOR_ATTRACTOR* maximised area covered by the swarm over time. This is shown in figure 4.8. The fact that the combinations space is relatively small (100 combinations) allows to show a combination per node. For each one, its weighted vector of the three morphogenesis variables is shown, as well as the corresponding color in the heat map of average area covered across all repetitions.

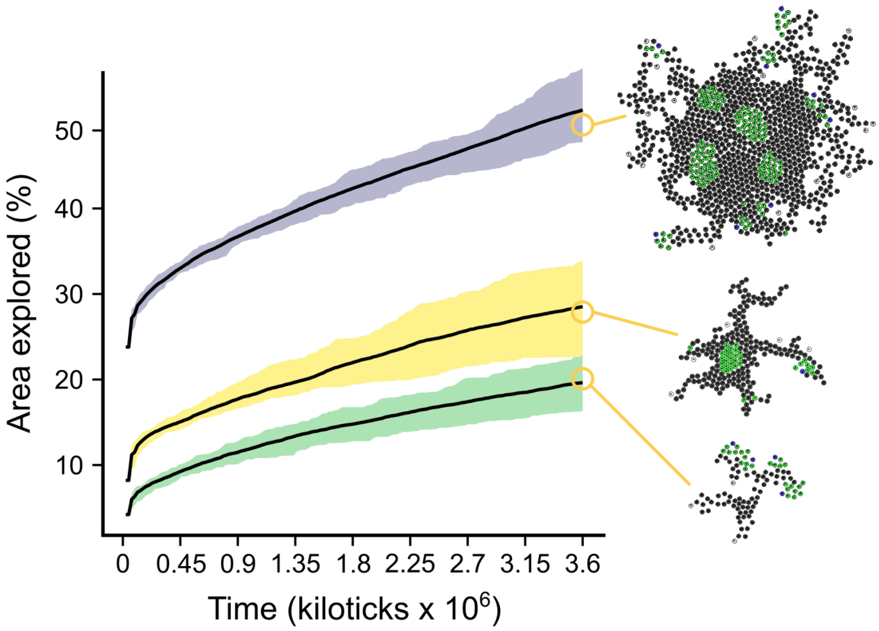


Figure 4.10: Scalability of the morphogenesis algorithm in simulation with combination {1, 5, 3} repeated 20 times. The average, minimum and maximum area covered is shown with 100 robots (bottom, green ribbon), 250 robots (middle, yellow ribbon) and 1000 robots (top, blue ribbon). One final shape is shown next to the corresponding curve.

#### 4.3.1.2 Real swarms

Three repetitions using combination {1, 5, 3} were performed using real swarms of 300 Kilobots to test whether morphogenesis occurred in real swarms. This combination was chosen arbitrarily from the three best combinations with no proved statistical difference. Robots were arranged in the same conditions as in simulation, i.e. in a circular initial shape at the centre of the arena. Initially, the swarms covered 6% of the whole area, approximately. Experiments were run for 3 hours and a half given battery autonomy. A similar performance can be seen across the three runs. Area covered increased to as much as twice the size of the swarms by the end of experiments. This shows that the algorithm can also grow shapes in real robots. Results of the area covered by the swarms over time are shown in figure 4.9.

### 4.3.2 Scalability

Scalability of the algorithm was also studied in simulation. Combination {1, 5, 3} was repeated 20 times with 100 and 1000 robots, and different random seeds, during a length of time equivalent to 8 hours in real time. Those simulations were compared with the results from the same combination using 250 robots. The curves present a similar behaviour over time, meaning that the inclusion of more robots does not negatively affect the performance of the algorithm, hence it is scalable. Results are shown in figure 4.10.

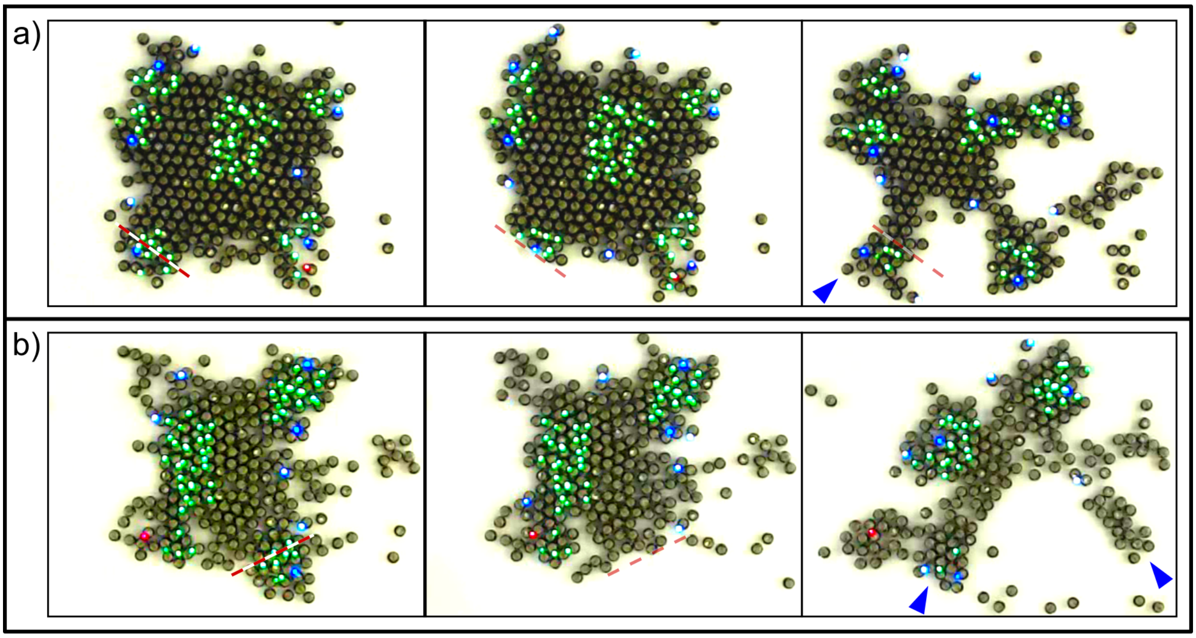


Figure 4.11: Robustness to damage in the controllable morphogenesis algorithm. *a)* Half a cluster is removed (dashed line), and regrows. *b)* One entire cluster is removed (dashed line), and branches grow on both sides.

### 4.3.3 Response to damage

Two additional experiments were performed to show the robustness of the algorithm to damage. For these experiments, the self-organised transition was improved (window length was reduced to one minute, and robots could also transit to the migration process in case they had at least two neighbours already in that process). In the first experiment, part of a branch was removed after one hour from the beginning of the experiment, leaving only a few robots in the cluster of that branch. In the second experiment, a whole branch and cluster were removed after one hour and a half. The robots manually taken from the swarm were completely removed from both experiments. As seen in figure 4.11a, the branch regrew completely and extended its length by the end of the experiment (about 3 hours and a half in total). In the second experiment shown in figure 4.11b, the branch did not grow back. Instead, branches on both sides of the cut could develop further after the same amount of time as in the first experiment. The result of these experiments show that the proposed morphogenesis algorithm is able to recover from damage, either by regrowing broken branches or reallocating more robots to the vicinity.

### 4.3.4 Response to obstacles

Finally, two extra simulations with swarms initialised with different random seeds in a scenario with the same obstacle below them were performed. The obstacle was thick enough to avoid robots communicating through it. In both cases, robots moved around the obstacle by surrounding it

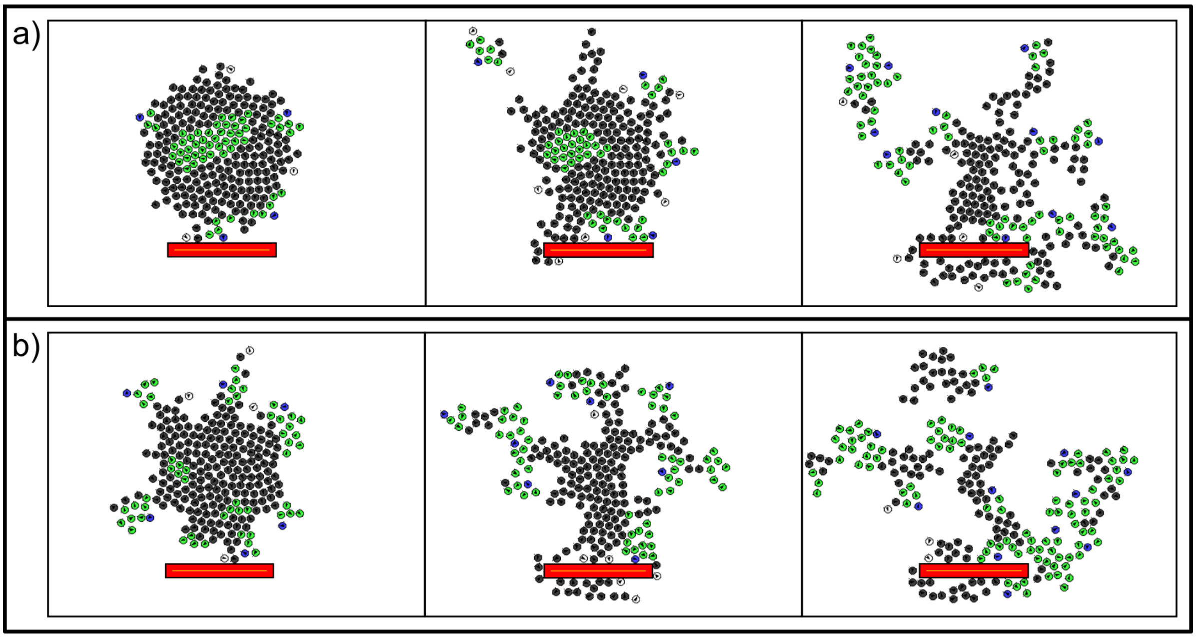


Figure 4.12: Response to obstacles in the controllable morphogenesis algorithm. Two different swarms ($a$ and $b$) get around an obstacle situated below them.

and reconnecting with the clusters in the growing branches on the right-hand side of the obstacle. This shows that swarms were able to adapt and continue exploring in the presence of an obstacle. Results can be seen in figure 4.12.

## 4.4 Discussion

Even though the area covered is similar across the six best variable combinations in simulation, shapes are qualitatively different. Each morphogenesis variable has a different impact. However, it is the combined effect of the three of them that produces different shapes. To understand such effect, simulations were done using the extreme values for each variable (for *MAX_HOPS*, a value of one instead of zero was used). Same seed was used across simulations. An analysis of the swarm morphologies was done using two different morphometrics to measure the effect of the variables quantitatively. Morphometrics were Shape Index and the minimum number of characterizing points, as described in §3.3.3. The morphogenesis variables that control the morphogenesis process produce greater richness of shapes compared to the morphogenesis algorithm from last chapter, as shown in figure 4.13. This is precisely what I mean by controllability in this chapter: as a result of the morphogenesis variables controlling the local gradients, there is a higher level of control on the type of shapes that emerge in the swarm with respect to the morphogenesis algorithm developed in the previous chapter. For example, combination *{MAX_HOPS = 4, MIN_TO_STOP = 1, MIN_FOR_ATTRACTOR = 1}* produces stubby shapes, whereas combination *{MAX_HOPS = 1, MIN_TO_STOP = 9, MIN_FOR_ATTRACTOR = 1}* produces shapes that are thin and long. This

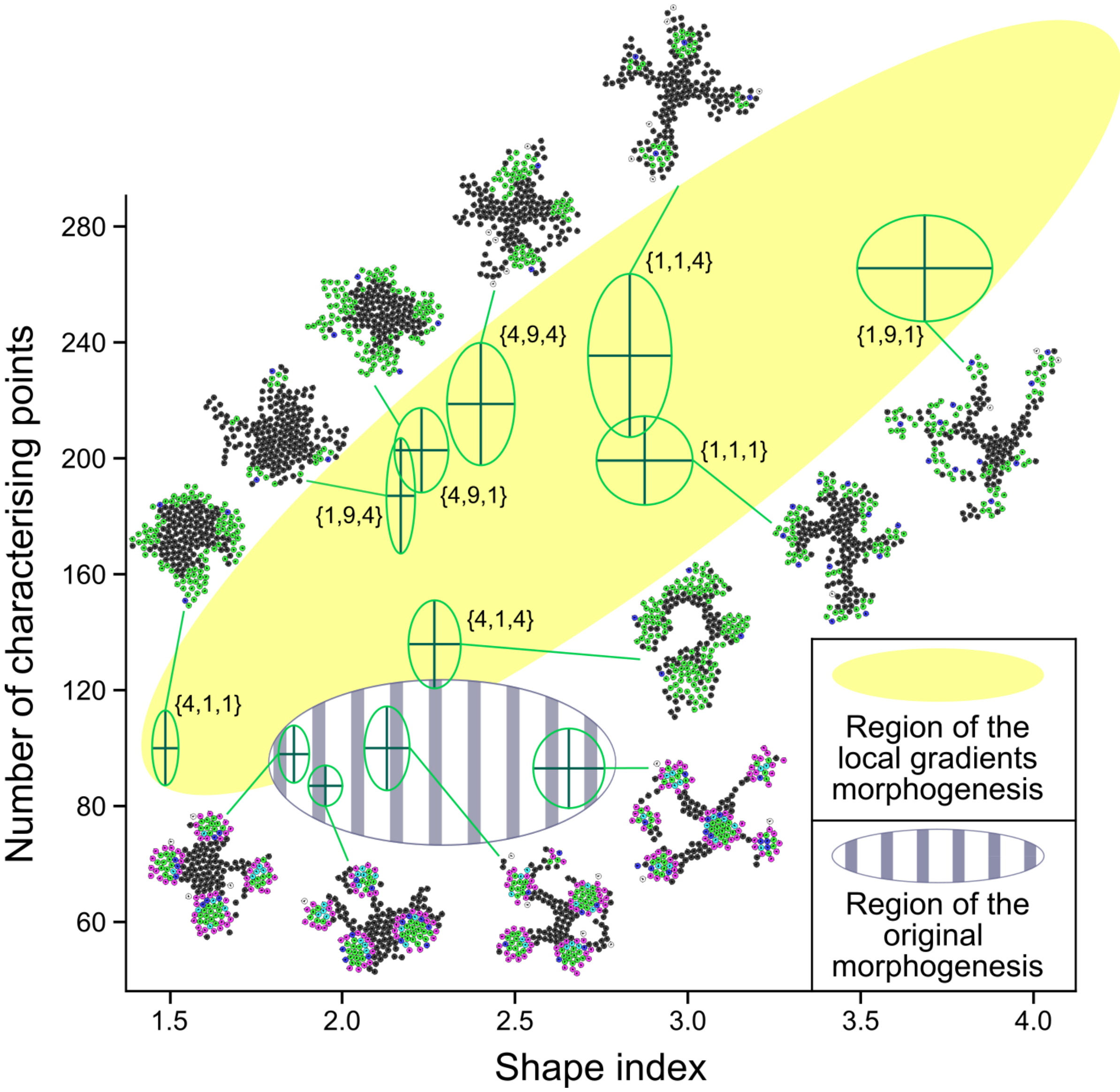


Figure 4.13: Morphospace produced by different combinations of morphogenesis variables (labelled as {*MAX_HOPS, MIN_TO_STOP, MIN_FOR_ATTRACTOR*}) and the original morphogenesis algorithm from the previous chapter. Ellipses for each combination of 20 repetitions show the standard deviation in the two metrics (Shape Index and number of characterising points). Ellipses are centred at the median of the results for each combination. Snapshots show the final shape of the closest repetition to the median metrics for each combination.

could be advantageous in situations where the swarm has to adapt to the environment, e.g. when performing collective area coverage. By modifying the morphogenesis variables, the swarm could produce a whole range of different growth depending if wider/thicker or further/thinner reach is required for the task. However, it is important to acknowledge that the algorithm presented in this chapter achieves less controllability than those that use explicit global blueprints in the form of a map of the shape to create (e.g. [Rubenstein et al., 2014b; Gauci et al., 2018]), meaning that swarms cannot grow a specific shape provided by the user. This might not be a disadvantage for certain tasks. For example, if the swarm was completely blocked by an obstacle, the algorithm presented in this chapter could be easily extended to allow robots to create new clusters of attracting robots during the migration process where needed to grow new exploration branches, hence modifying its shape (as opposed to having several predefined shapes that the swarm could grow).

With respect to the effect of the morphogenesis variables, a higher number of *MAX_HOPS* does not produce branching out as effectively. The reason is the size of the clusters. If the hops threshold is high, one source can include many robots in the cluster. The edge then gets saturated with robots in a cluster, causing a wider expansion until the point where no robots on the edge move because they belong to a cluster. As a consequence, the lower *MAX_HOPS*, the better for branching out. In the case when *MAX_HOPS* is zero, sources of the signal are the only ones in their own cluster. Therefore, moving robots will stop near sources, depending on the other values. This causes a branching out effect as well, as seen in figure 4.7. Another crucial aspect regarding *MAX_HOPS* is that only the robots inside a cluster are allowed to belong to it. If instead all robots meeting the *MAX_HOPS* criteria were allowed, this would have a similar effect to the behaviour seen when *MAX_HOPS* is high. Eventually, more and more robots would become part of a cluster, hence blocking the robots on the edge of the swarm from moving.

A low value of *MAX_HOPS* alone does imply branching out. Due to communication noise, for example, a robot might stop next to a cluster independently of *MIN_TO_STOP*. If *MIN_FOR_ATTRACTOR* is low enough, the robot will become part of the cluster. The lower this threshold, the more likely it is that stopping robots will add to clusters, thus, helping morphogenesis. If the value is high (or *MIN_FOR_ATTRACTOR* > *MIN_TO_STOP*), robots might stop but not join the cluster because there would not be enough attracting neighbours. This means that less branch growth would be achieved, hence, stopping morphogenesis. As a consequence, the lower *MIN_FOR_ATTRACTOR*, the better for branching out.

When *MAX_HOPS* and *MIN_FOR_ATTRACTOR* are low but *MIN_TO_STOP* is high, branches are thinner and longer. This has to do with the tendency of this combination to create bifurcations. With these conditions, sources tend to appear on the sides of the cluster when it reaches a certain size. A high number of sources can also be seen when the three variables are low. However, the behaviour is slightly different. In this case, robots in the middle of a cluster may eventually leave it after a bifurcation. But as soon as those robots move again, they will stop because of their attracting neighbours and the low threshold for the stopping condition. Therefore, lost parts of the cluster tend to be recovered straight away, causing a wider growth. When *MIN_TO_STOP* is high, the robots in the middle leaving the cluster will orbit a longer distance around it, helping bifurcation. As a consequence, the higher *MIN_TO_STOP*, the better for branching out. The three conclusions discussed in this subsection match the results shown in figure 4.7.

In the real swarms, growth was produced in a similar fashion across the three runs as a result of the morphogenesis variables. After patterning created four clusters on the edge of the swarm, four branches of similar thickness grew. Results show that the area covered increased over time repeatedly in the three runs. This confirms the morphogenesis process can also be controlled in large swarms of simple, real robots. However, swarms covered less area in the same amount of time compared to simulation, as seen in the middle curve from figure 4.10. This shows that there is still a reality gap between simulation and the real Kilobots. Due to their very noisy behaviour (especially when it comes to motion), it is hard to model them accurately in simulation. The addition of noise helps crossing such gap [Jakobi et al., 1995], but it does not manage to represent reality completely. A more accurate noise model will be considered in future work.

Swarms have been shown to be able to recover from damage or to continue growing even in the event of losing part of the robots. In the case of part of a cluster being removed (figure 4.11a, the branch could regrow because the damaged cluster could still attract other robots that eventually became part of the cluster and restored it. On the contrary, if the whole cluster disappeared (figure 4.11b, it could not grow back. Instead, such area in the swarm was freed up so that robots on the edge could travel to nearby clusters in both directions. This produced more growth in other branches. A mechanism to run patterning again when the swarm loses a certain number of clusters could help recover from more substantial damage, such as losing all branches. This will be studied in future work, as well as a mechanism to prevent the swarm from splitting.

Finally, swarms were also tested for their ability to get around a simple, static obstacle, as seen in figure 4.12. In both experiments, as soon as the swarms reached the obstacle, robots split into two branches. On the right-hand side of the obstacle there was a branch made of a cluster of attracting robots growing thanks to the robots that could arrive from its further right (the obstacle prevented robots from arriving from the left-hand side of that branch). The other part of the split was made of robots migrating around the swarm towards the left-hand side because of the obstacle again. However, the behaviour of those robots was to orbit around the perimeter of the obstacle (a behaviour emerging from edge following mechanism that makes them orbit around their nearest neighbour). When enough robots had positioned around the perimeter of the obstacle, subsequent robots could join the other part of the split (the growing branch with a cluster of attracting robots), hence successfully getting around the obstacle. It is important to acknowledge that the obstacles placed in those two experiments were very simple. To completely characterise the ability of the swarms to get around obstacles, more complex shapes (e.g. thicker obstacles, or other shapes apart from a line) are required. This is left as future work.

## 4.5 Towards functional morphogenesis

To realise the goal of morphogenetic engineering and to ultimately deploy robot swarm morphogenesis in real-world applications, the next step from controllability is functionality. Some of the work described in §2.2.1 has shown functionality in the form of object transport, target tracking or shortest-path creation [Liu et al., 2017; Oh et al., 2018; O'Grady et al., 2012; O'Grady et al., 2010; Groß et al., 2006b].

In this section, an extension to the controllable morphogenesis algorithm—whereby the swarm is able to explore the environment to find objects of potential interest and find the shortest path between them—, is explained and tested in simulation. Inspiration comes from the complex behaviours seen in the unicellular organism *Physarum polycephalum*, commonly known as *true slime mould*. This organism is composed of one cell and multiple nuclei in the plasmodium state of its life cycle, and it is able to modify its shape to forage [Alvarado and Stephenson, 2017], as shown in figure 4.14. It explores the environment by branching out protoplasmic tubes and using chemotaxis when attracted or repelled to chemical gradients in the environment [Ueda et al., 1976; Durham and Ridgway, 1976]. Despite the lack of a nervous system, *Physarum polycephalum* shows a wide range of complex behaviours such as spatial memory [Reid et al., 2012], learning by habituation to stimuli [Boisseau et al., 2016], multi-objective decision making [Beekman and Latty, 2015; Dussutour et al., 2010], maze solving [Nakagaki et al., 2000], fault-tolerant network planning [Tero et al., 2010] or computation [Adamatzky, 2016]. Undoubtedly, these behaviours would be very useful for a swarm of robots growing shapes to explore an environment in search and rescue operations, for example.

The general idea behind the functional morphogenesis algorithm is that robots perform controllable morphogenesis to explore while looking for objects placed in the environment that represent targets to find. I call these here *objects of potential interest* (OPI). In a real application scenario, OPI could be a casualty in a building on fire, a crack in a water pipe, or cancer cells. When robots find OPI, they connect them through the shortest path between them, given the swarm spatial configuration. They do so by lighting up the shortest path with their LEDs. This way, the swarm offers swarm-guided navigation to other robots or human users [Brambilla et al., 2013]. Robots in the shortest path between two OPI remain stationary to avoid breaking the connection between them, whereas the rest of robots continue exploring the environment to find more OPI or optimal paths between the OPI already found, as in *Physarum polycephalum*.

### 4.5.1 Methodology

OPI are represented as circular blue regions of constant light (not a gradient) in simulation because Kilobots are able to sense ambient light above them. The aim of the swarm is to grow a shape to explore and find the projected OPI in the environment to ultimately have the robots in the shortest path signalling it by lighting up their LEDs. To achieve this, algorithm 3 was

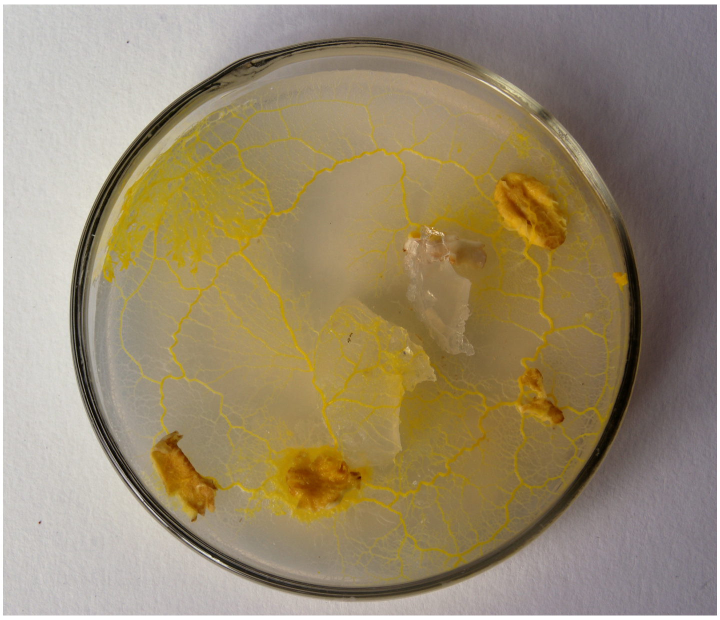

Figure 4.14: *Physarum polycephalum* in a growth medium. *Credit: Rob Cruickshank. Licensed under CC BY 2.0.*

extended. In this extension, robots store the OPI they find and/or receive from neighbours in an constant-size array, as well as the paths that are created by joining two OPI. Every timestep during morphogenesis, all robots check if they have found an OPI by checking their ambient light, no matter the state they are in. To increase robustness, they have to sense the OPI for at least a certain number of consecutive times, defined by a constant *MIN_TIMES_LIGHT_DETECTED* (which is equal to 10 in the simulation). If they surpass this threshold, they store the OPI in their OPI table with a random 2-byte ID, and start transmitting it to neighbours with a hop count equal to zero. OPI messages contain the ID of the OPI (2 bytes) and the number of hops to reach it (1 byte). When a robot receives a message with an OPI, it also stores it in its OPI table and transmits it to neighbours. Robots always keep the version of the OPI with a lowest number of hops increased by one hop, as was the case for the local gradients algorithm. This way, all robots store the lowest number of hops to reach a particular OPI that they have sensed or received from a neighbour. The number of OPI they can store is limited by the constant *MAX_OPI*. In case they receive more OPI than the maximum number they can store, they will replace the furthest one, i.e. the one with a higher number of hops to the robot that has sensed the OPI.

When a robot senses an OPI directly, it creates a path between that OPI and the rest of OPI in its table, with a number of hops equal to the corresponding number of hops per each OPI, as stored in the OPI table. These paths are indeed the shortest paths between the sensed OPI and the rest of OPI, as the robot is measuring one end of the path directly—therefore, zero hops to that OPI—, and it can reach the other end of the paths in the number of hops stored in its OPI table—which are the lowest number of hops to reach each particular OPI. Robots also transmit paths messages to neighbours for them to store in their corresponding paths tables. Paths messages contain the two IDs of the OPI (4 bytes), the number of hops of the shortest path (1 byte) and a timestamp with the most recent time the path was updated for robots to update the path appropriately (4 bytes). A timestamp is needed for paths because hops are fixed for them, as opposed to storing the lowest number of hops to reach an OPI. Therefore, robots that create path messages are the only ones allowed to update the timestamp. This prevents misinformation being spread by the rest of the robots in the swarm. Moreover, in the same way as OPI tables, the number of paths that a robot can store in its paths table is limited by another constant, which I decided to be the same as *MAX_OPI* for memory reasons. The effect of different table sizes is explored in §4.5.2.1. It is worth mentioning that robots sensing an OPI directly do not run the local-gradient based migration but stay still, unless they stop sensing the OPI.

Finally, when a robot receives a message with the shortest path between two OPI, it checks whether it belongs to it by i) checking if the two OPI are in its OPI table (otherwise, it would not be able to reach them), and if they are, ii) whether the sum of the hops to reach both OPI from this particular robot—which is the sum of the hops stored in its OPI table—, equal to the number of hops of the shortest path received. If this happens a certain number of consecutive times defined by the constant *MIN_TIMES_SHORTEST_PATH_DETECTED* (for the sake of robustness; 10 times in the simulation), this robot is in the shortest path between the two OPI. As a result, it lights up its LED to signal this condition. Furthermore, it also stays still by not running the migration process, unless it does not belong to the shortest path anymore. Robots that do not belong to any shortest path are the ones that continue exploring the environment through migration.

If robots have OPI and/or paths to send apart from local gradients information, they have to alternate among types of messages—Kilobot messages are constrained to 9 bytes. To counteract this, they switch the type of message they send every *FREQUENCY_NEW_MSG* seconds (set to 1s in the simulations). They start by sending a local-gradients message. Then, they switch to an OPI message, transmitting the first OPI in their table during *FREQUENCY_NEW_MSG* seconds. After this, they transmit their local-gradients message again. When they have done this for *FREQUENCY_NEW_MSG* seconds, they transmit the second OPI they have in their OPI table. Then they go back to the local-gradients message again, and they repeat this sequence until all OPI have been sent. At this point, they repeat the same sequence with all paths that they have in their paths table, if they have any. When these are done, they start with the first OPI, and so on. It is then very important to adjust the purging frequency of OPI and paths to the frequency that these messages are sent to neighbours. On the one hand, if purging frequency is low, updates would take a long time to reach robots. On the other hand, if purging frequency is high, information might be lost. A study on the best values and/or different strategies should be done in the future to optimise information flow across the swarm. A diagram with an overview of the functional morphogenesis, as explained above, is shown in figure 4.15. The pseudo-code of this algorithm is shown in algorithm 4.

**Algorithm 4:** Pseudo-code of functional morphogenesis

```
// Initialisation: random molecule concentrations, robot not moving
setup()
// loop() function
while TRUE do
    /* It processes received messages, and updates neighbours' tables and running
       averages of neighbours and neighbours of neighbours                          */
    receiveInputs()
    // PATTERNING
    if kiloticks < PATTERNING_TIME then
        /* Concentration of molecules U and V is updated based on the linear model for
           reaction-diffusion                                                        */
        reactionDiffusion()
        patternIsStable ← FALSE
    // SELF-ORGANISED TRANSITION
    else if not patternIsStable then
        /* It works out the cumulative change in concentration of molecule U during a
           window of w minutes                                                       */
        sum_u ← cumulativeChange()
        if sum_u ≤ 1 then
            patternIsStable ← TRUE
            if u ≥ 3 then
                state ← ATTRACTING_ROBOT
            else
                state ← NON_ATTRACTING_ROBOT
            end if
        end if
    // MIGRATION
    else
        /* It check whether the robot is sensing an OPI and/or is in the shortest path
           between two of them                                                       */
        inShortestPath ← checkShortestPath()
        if inShortestPath then
            /* If the robot is sensing an OPI directly, it creates/updates the path
               between it and all the OPI received from neighbours.                  */
            updatePaths()
        else
            /* Non attracting robots on the edge orbit the swarm until they find clusters
               of attracting robots. The latter maintain them through local gradients.
               Robots also try to recover if they get lost.                          */
            localGradientsMigration()
        end if
    end if
    // It shows the corresponding LED colour depending on the process
    showColour()
    // It computes a new, unique, local ID in case of clash
    checkLocalID()
    /* It removes entries from the table, OPI and paths older than s seconds         */
    purgeEntries()
    /* It updates the message that it is sent depending on the process. During
       migration, it alternates between local gradients and OPI/paths.               */
    updateMessage()
end while
```

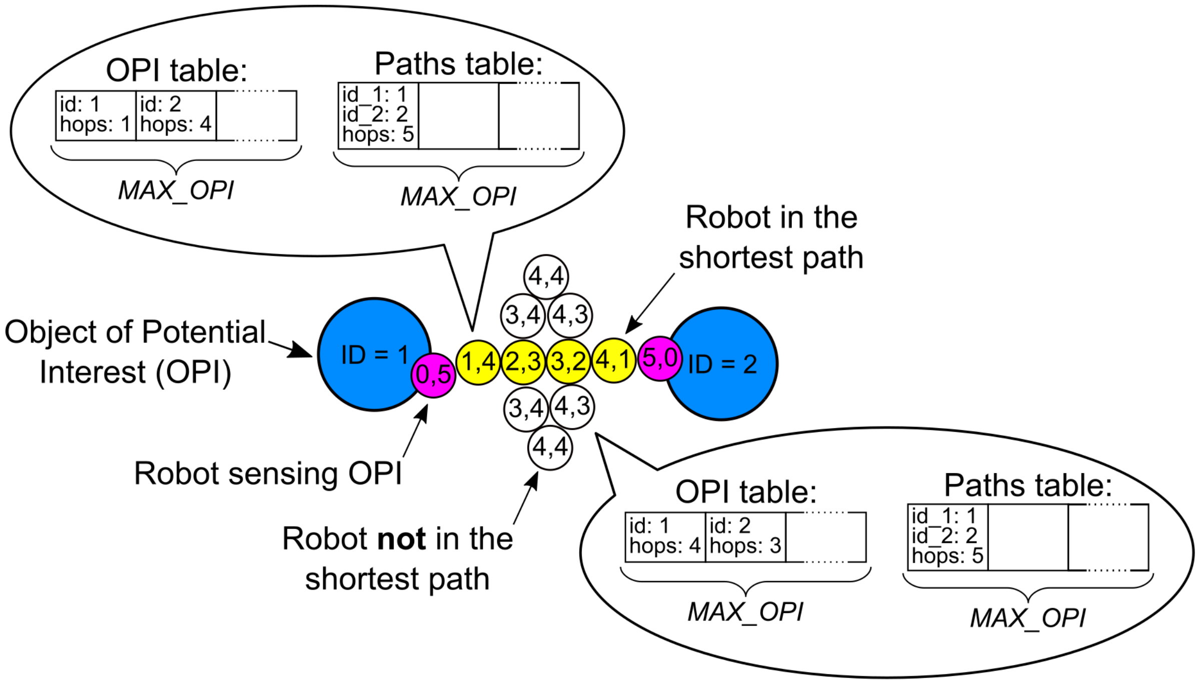


Figure 4.15: Diagram of a small swarm of robots doing functional morphogenesis.

### 4.5.2 Results in simulation

The extended algorithm for functionality was tested with a swarm of 250 simulated Kilobots in a very simple scenario with two OPI at both sides of the swarm. Swarm grew a shape to explore the environment, found and connected the two OPI while continuing exploration with two other branches. Snapshots of the whole process of patterning, migration, and shortest path creation can be seen in figure 4.16. This is an important step towards functional morphogenesis. Possible extensions and more testing scenarios are explored in §4.5.3.

#### 4.5.2.1 Effect of different table sizes

A basic study in simulation of the effect of different OPI and path tables sizes was done with swarms of 1000 static Kilobots (no patterning and no migration), and 5 OPI in the environment directly inside the swarms. Concretely, values of *MAX_OPI* equal to two, five, eight and eleven were tested, with the maximum number of OPI to store being the same as paths, as described above. When only two OPI are stored, the swarm finds the smallest paths between two OPI. This is in fact the minimum number of OPI that a robot must store to find a path, as paths are composed of two OPI. In the case of $MAX\_OPI = 5$, two more paths are formed from the OPI by the middle of the swarm to the other unconnected OPI at the top and right-hand side of the swarm. These are the next paths with the smallest length. That is the reason they appear from the middle of the swarm rather than from the sides. Another reason for not having a path

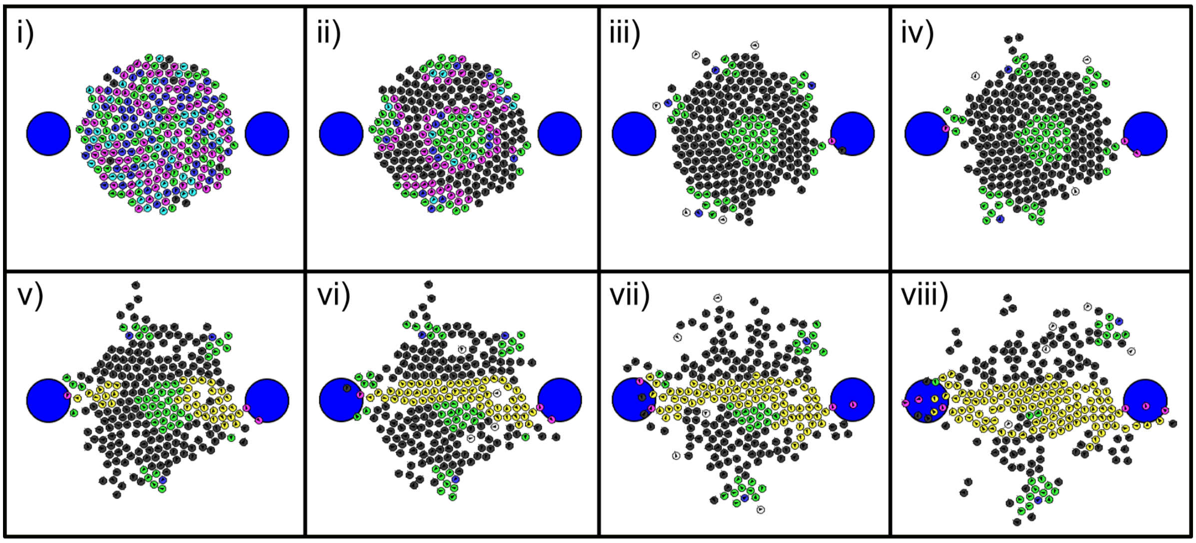


Figure 4.16: A swarm of 250 simulated Kilobots performing functional morphogenesis. *i)* Initial random concentration of molecules. *ii)* Stable pattern. *iii)* A robot finds an OPI at the right-hand side of the swarm. *iv)* A robot finds another OPI at the left-hand side of the swarm. *v)* Shortest path begins to emerge with robots lighting up in yellow. *vi)* The shortest path between OPI is completed. *vii-viii)* The swarm continues exploring the environment while updating the path.

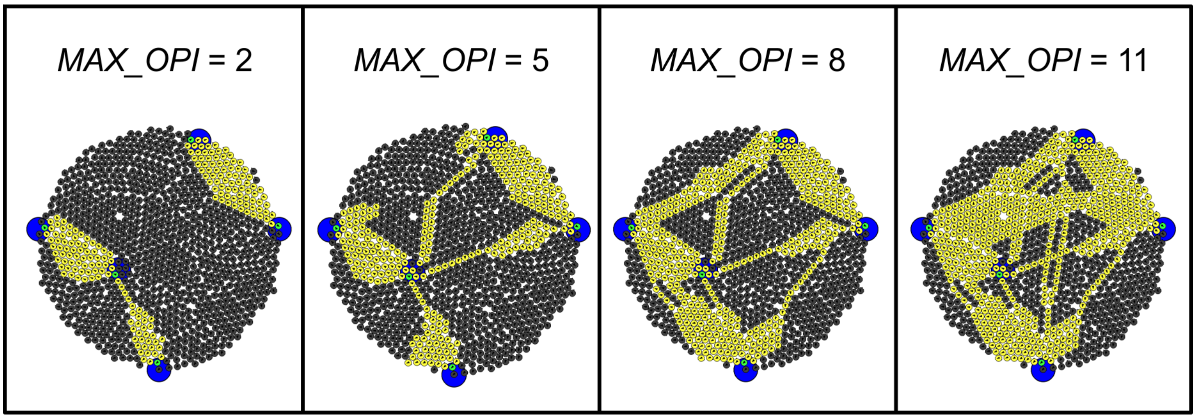


Figure 4.17: Effect of *MAX_OPI* in a static swarm of 1000 simulated Kilobots with 5 OPI in the environment.

between the top OPI and the left-hand side OPI is that robots store the smallest paths, even if by doing so they replace a longer path that they belong to. In this case, the path between the top OPI and the right-hand side OPI is smaller than the one that would be created between the top OPI and the left-hand side OPI. Therefore, robots in that hypothetical path store the four paths created by the middle OPI, plus the one from the top OPI to the right-hand side OPI. This situation is solved by increasing *MAX_OPI*. When $MAX\_OPI = 8$, that path is created, as well as the one from the bottom OPI to the right-hand side OPI. Finally, when $MAX\_OPI = 11$, all possible paths between OPI (in this example, the number is 9) appear—see figure 4.17.

Having a limited number of OPI and paths stored makes the algorithm scalable to the number of OPI in the environment as well as adaptable to memory requirements of the robots. However, this comes at the price of limiting the number of connections that are shown between paths. In the explained version of the algorithm, the bigger *MAX_OPI*, the more and longer paths between OPI are shown by the swarm. On the contrary, if *MAX_OPI* is low, only the smallest paths are shown between OPI. All in all, *MAX_OPI* controls how connected the graph is. In a real scenario, a small *MAX_OPI* might not be a problem, as not every possible connection might be needed for swarm-guided navigation. Instead, one could jump from OPI to OPI through their paths to discover the rest. When taking this approach in real-world applications, this parameter should be adjusted depending on the connectivity that is required by the particular application. Furthermore, a different number of OPI and paths to store should be studied, as well as different replacing strategies for OPI and paths. This remains an open line of research.

### 4.5.3 Extensions to the functional algorithm

The simplicity of the bottom-up approach proposed in this chapter in particular allows to add new rules to the robots or slightly modify the conditions in order to obtain new emergent behaviours relatively easy. For example, chemotaxis (gradient-based attracting/repelling behaviour of the branches) could be implemented if the rule governing when a robot stops and moves is modified. If priority is given to stopping when a moving robot detects a concentration of a (representation of a) chemical in the environment higher than the one which the source robot of the cluster is sensing, movement of the branch would in principle be directed upwards in the gradient. As far as repelling is concerned, robots could be allowed to move if they detected a repellent in the environment, even if they are part of a cluster. The expected behaviour would be branches moving outside repelling zones in the environment.

Another important addition to the algorithm is concerned with adaptability. In a real search and rescue scenario, adapting to unknown, dynamic environments is crucial. The functional algorithm could be extended for adaptability by making use of the first process, patterning, as many times as needed. Since the role of this process is to automatically define clusters of attracting robots, patterning could be triggered in case a branch is blocked by an obstacle. This could be automatically detected by the robots by adding a threshold for the time the cluster has had the same source, for example. Moreover, the values of the morphogenesis variables could be dynamically adjusted to the conditions of the environment.

When *Physarum polycephalum* finds a source of food, it engulfs it. As it adds more nutrients to the protoplasm, the organism grows in size, especially around the food. At the same time, the shortest path between found sources of food is reinforced, whereas branches belonging to the longest paths move backwards to finally disappear. This behaviour emerges when enough exploration time is given to the slime mould. However, the simple scenario with two OPI tested above did not allow to check this. In the future, more scenarios with a higher number of OPI in the environment, appearing and disappearing, and being blocked by obstacles, should be studied. This remains as an open line of research.

## 4.6 Concluding remarks

In this chapter, a new morphogenesis algorithm has been designed for large swarms of simple robots with minimal capabilities. For this, reaction-diffusion, migration and positional information have been combined to allow different interpretations to the pattern first established by reaction-diffusion, showing a higher degree of controllability compared to the morphogenesis algorithm developed in the previous chapter. In fact, the combination of both developmental theories has been hypothesised to happen in nature [Green and Sharpe, 2015]. By decoupling the patterning and the migrations processes, spots emerging from patterning could be controlled through local gradients [Mamei et al., 2004] during the migration process. As a result, robots self-organised to grow shapes while maintaining the communication network without the need for a map of the shape, a coordinate system or preprogrammed seed robots. Controllability was achieved by having three different morphogenesis variables influencing the local gradients. The best values for these morphogenesis variables have been identified by performing over 2000 simulations. Results have shown the rich morphospace that these variables are able to produce. This has the potential to enable adaptive growth dynamically if required, i.e. robot swarms could grow differently to adapt to changes in the environment by modifying the morphogenesis variables. This could be particularly useful for real-world applications.

One of the best combinations found was tested 3 times on real swarms of 300 Kilobots. This combination produced similar results repeatedly across the three runs. Furthermore, 2 experiments with real swarms of 300 Kilobots were conducted to show that the algorithm is robust to damage. In one experiment, part of a branch was cut off after one hour of morphogenesis, and managed to grow back completely. In another experiment, a complete branch was removed, resulting in longer growth of the branches on both sides of the missing branch. Extra simulations with different number of robots (100, 250 and 1000) have shown that the algorithm is scalable, thanks to being only based on local information. Swarms grew a shape in a similar manner independently of the number of robots. Two more simulations showed how the swarms could continue growing in the presence of an obstacle.

The work carried out in this chapter represents an example of bottom-up, self-organised, controllable, scalable, adaptable and robust morphogenesis algorithm tested on large swarms of real robots, getting a step closer to the goals of morphogenetic engineering. I have also shown in simulation how the algorithm can be extended for functional morphogenesis, i.e. a swarm of robots growing shapes to explore an environment, finding and connecting objects of potential interest through the shortest paths, which are signalled to the user for swarm-guided navigation [Brambilla et al., 2013]. This could indeed be very useful in disaster scenarios [Hauert et al., 2009; Murphy et al., 2008], where a swarm of robots is released to explore the environment and deploy exit routes by creating a communication/visual chain to guide other robots or humans, for example.

The following items should be considered for future work:

- In the controllable and functional algorithms explained in this chapter (as well as in the original morphogenesis algorithm from the previous chapter), there is no mechanism in place to prevent the swarm from splitting into islands, therefore breaking connectivity. The $\beta$-algorithm proposed by Nembrini and Winfield [2012] might be an option to use as a way of restoring connectivity in case the swarm splits up.

- In terms of the functional morphogenesis algorithm, experiments have only been made in simulation. An extension would be to adapt the algorithm to real Kilobots facing environments with a different number of OPI, obstacles and dynamic conditions, as described in §4.5.3. For this, the optimum values of the different parameters involved in the process must be found.

- As different initial configurations of the swarm might cause more robots to be at the edge of the swarm (hence, having more robots moving at the same time), performance of the algorithms might be improved. Therefore, a study on the best initial configuration should be done to understand how they affect performance in terms of area covered over time.

# Chapter 5

## What society wants with robot swarms

The previous chapters focussed on completely self-organised, controllable and functional morphogenesis in large robot swarms of real, simple robots. Given that the previous research is still in the early stages, now is the perfect time to review its application and ethical implications with potential users and stakeholders in particular and with the general public, if we want to unleash the predicted economic and societal benefits of swarm robotics this decade. This chapter comprises two studies that consider the inclusion of society (users and the general public) in swarm robotics research. The first study was done with fire brigades through focus groups following the framework of mutual shaping, and the second study was done with the general public through gamification in the form of an educational escape room and a group discussion session. These studies are the first studies of their kind (to the best of my knowledge) that formally engage users and the general public by understanding their attitudes, hopes, concerns and requirements about the topic of swarm robotics in particular.

This chapter begins with a summary of related work in human-robot interaction and human-swarm interaction. Then, the study with fire brigades is presented, showing the results of 3 focus groups with a total of 23 participants, where their attitudes, concerns and requirements of robot swarms were identified. This study is followed by the study with the general public in the second half of the chapter, where a total of 52 people participated to learn about swarm robotics and express their attitudes, hopes and fears about this particular technology. Finally, the main findings of both studies are summarised as concluding remarks at the end of this chapter.

The study with fire brigades presented in this chapter has been published in the following peer-reviewed journal:

- Carrillo-Zapata, D., Milner, E., Hird, J., Tzoumas, G., Vardanega, P., Sooriyabandara, M., Giuliani, M., Winfield, A. F. T., & Hauert, S. (2020). Mutual Shaping in Swarm Robotics:

User Studies in Fire and Rescue, Storage Organization, and Bridge Inspection. Frontiers in Robotics and AI, 7(53).

– **Personal contribution**: I designed the study, recruited the participants and facilitated the study with personnel from fire & rescue services, analysed transcripts and questionnaires, and substantially contributed to the writing of the manuscript as co-first author.

Most parts of §5.2and some parts of §2.3 and §5.4 have been reproduced verbatim from the previous paper.

## 5.1 Introduction

One of the possible applications of the research carried out in chapters 3 and 4 is search and rescue, where robot swarms are deployed to grow shapes to explore a disaster environment (e.g. a building on fire) and locate casualties, the source of the fire, etc. As described in chapter 2, researchers have used interviews, questionnaires and workshops to engage potential users such as fire brigades. However, researchers have mainly focused on understanding their response to a pre-existing robotic prototype [Penders et al., 2011; Yanco et al., 2006; Driewer et al., 2005], or their response from a psychophysiological point of view [Podevijn et al., 2018, 2016b], with barely any study considering the ethical aspects of robots being used in search and rescue [Harbers et al., 2017], much less concerning swarm robotics. Because these studies have been done in the form of ethnographic and user-centered studies, there is a lack of inclusion of users in the research and development process of robot swarms. The lack of involvement of users from an ethical perspective, and the growing negative public perception driven by science fiction and hyped headlines [Hamann, 2018], may result in a decrease in acceptance and trust, eventually slowing down or even putting at risk the predicted advances and impact of swarm robotics technology during the decade 2020–2030 [Yang et al., 2018]. To counteract this, participatory approaches such as mutual shaping [Boczkowski, 1999], where researchers and participants discuss practical and ethical aspects of a particular technology in a way that has a bidirectional impact on both of them from the very beginning [Winfield and Jirotka, 2018], could be used.

In fields such as socially assistive robotics, participatory studies following the mutual shaping methodology are much more common. Of particular interest is the study carried out by Winkle et al. [2019], where authors organised several focus groups with therapists, and showed that their mutual shaping methodology i) positively influenced participants acceptance, and ii) provided guidance to researchers to maximise benefit of the technology and reduce negative consequences when deployed. Therefore, their methodology was used to structure the study with fire brigades, which addressed the following question: What are the main needs, attitudes and concerns from fire brigades with respect to robot swarms being used to assist them in fire & rescue missions?

Three focus groups with different roles within fire brigades were organised. This study involved a total of 23 participants, and allowed us to assess aspects such as their attitudes towards working alongside robot swarms, their preferred number of robots to use, their mode of operation, or their thoughts towards scenarios of applications arising from the research done in chapters 3 and 4.

Regarding engagement of the general public in scientific and technological topics, several approaches have been proposed so far, ranging from generating public dialogue as the main goal [Bucchi and Trench, 2014], to the inclusion of the public in *doing* science (e.g. citizens science) [Shirk et al., 2012]. In particular to swarm robotics, engagement has mainly focused on educational aspects, i.e. teaching about the principles of the technology, as described in §2.3.2. For a successful engagement experience, previous research has shown that having fun is key for the audience to learn and become more interested in STEM [Carrillo-Zapata et al., 2020]. As described in §2.3.2, gamification was used by Becker et al. [2014] in the form of an online game to teach players about robot swarms through a series of manipulation challenges, and to teach researchers how players achieved better performance. However, neither impact on learning nor attitudes or concerns from players were measured in this study. Hence, there is a gap in understanding how the general public perceives swarm robotics technology.

To bridge the previous gap, the second study addressed the following question: What are the main attitudes and concerns from the general public about robot swarms being applied to solve real-world problems? Given that gamification has the potential to improve learning [Dicheva et al., 2015], and to encourage long-term reflections [Nicholson, 2015], I decided to design an educational escape room to engage the general public in the topic of swarm robotics morphogenesis in a more interactive way.

Among gamification approaches, educational escape rooms have emerged as powerful engagement tools due to the growing market and worldwide popularity of commercial escape rooms [Clarke et al., 2017], and their potential to create memorable learning experiences [Nicholson, 2018]. As defined by Nicholson [2016], escape rooms are *"live-action team-based games where players discover clues, solve puzzles, and accomplish tasks in one or more rooms in order to accomplish a specific goal (usually escaping from the room) in a limited amount of time"*. Many educational escape rooms have been proposed so far [Fotaris and Mastoras, 2019] to teach subjects such as mathematics [Arnal et al., 2019], physics [Vörös and Sárközi, 2017], computer science [Ho, 2018; López-Pernas et al., 2019], or robotics [Giang et al., 2018]. The educational escape room and accompanying group session engaged a total of 52 people, allowing us to assess aspects such as their attitudes towards being assisted by robot swarms in their work/home, the benefits versus risks of this technology, their preferred uses of robot swarms, or the actions that they thought researchers should be doing to successfully deploy robot swarms in the real world.

## 5.2 A mutual shaping study with fire and rescue services

This study consisted of three focus groups with fire brigades. Participants ranged from firefighters to personnel from Research & Development departments. The diversity in participants allowed to have opinions from firefighters with real firefighting and rescuing experience, as well as from the people working in more technical fields related to development of processes.

### 5.2.1 Methodology

The mutual shaping structure successfully applied by Winkle et al. [2019] was used to structure this study. Authors propose to split up mutual shaping sessions in three main parts:

1. **Pre-demonstration Discussion**: to understand participants' initial ideas on the topic before being given information.

2. **Project Presentation and Robot Demonstrations**: to introduce participants to the topic of the session by giving an overview of the state of the art, aims of the project, an explanation of the topic and (perhaps) a robot demonstration.

3. **Post-demonstration Discussion**: for participants to give their opinions to researchers about the topic as well as their requirements to advance in the development of the particular technology in discussion.

I adapted their methodology to the topic of swarm robotics. Focus-group-style sessions were chosen to have teams with different roles discussing the topics and contrasting opinions during the same session, as opposed to interviewing firefighters individually. Focus groups were designed to have the following structure:

1. **Art of their profession.** This first part consisted of an initial discussion with participants to understand what their profession involves, as well as their initial attitudes towards the use of robots for firefighting and rescuing. The following questions were verbally asked to participants to understand their jobs and processes:

    a) What is your area of work?

    b) What are the typical tasks in your job?

    c) What tools do you currently use?

    d) What technologies have been introduced into firefighting and rescuing?

    e) What do you think about using robots as a tool to help in your job?

    f) In which firefighting/rescuing tasks would robots be most useful?

2. **Introduction to swarm robotics and possible scenarios of application.** In this part, information was given to participants to introduce them to swarm robotics. I used presentation slides on a computer to present the following:

    a) **Current state of robots for firefighting/rescuing.** An overview of recent approaches suggested by the research community was given to participants to show the current state of the art of robots for firefighting/rescuing, as well as their limitations (e.g. bulky robots that may become an obstacle).

    b) **International competitions for multiple robots**. The ERL Emergency Service Robots competition held in Italy in 2017 [Winfield et al., 2016], and the Swarm and Search AI 2019 Fire Hack held simultaneously in the UK and US in 2019 were described to show international efforts in promoting research for robots assisting in fire and rescue operations.

    c) **Explanation of swarm robotics**. A video explaining the main concepts of swarm robotics (a large number of robots performing a collective behaviour in a decentralised fashion without any leader, inspired by natural swarms) was shown to participants [SciShow, 2017].

    d) **Possible scenarios of application.** Four different scenarios were shown one by one using presentation slides on a computer. These scenarios are related to morphogenesis in the sense that swarms spatially organise to accomplish the different tasks shown in figure 5.1. Each scenario is described below:

        i. **Indoor sensing**. This scenario depicts a swarm of (aerial) robots exploring a building on fire and creating a heat map of the building plus the location of firefighters and hazards.

        ii. **Indoor communication paths and exit routes**. A swarm of (aerial) robots is shown inside a building. In this case, the swarm has located a casualty in one part of the room and it is maintaining a communication link between the entrance of the building (representing the firefighters base station) and the casualty. In addition, robots in the shortest path between the entrance and the casualty are lighting up to show they are on the exit route.

        iii. **Indoor fire extinguishing**. In this scenario, a swarm of robots (ground, moving balls designed by one of the authors, and similar to Sphero$^{\text{TM}}$ robots, shown in figure 5.2) is released inside a building on fire. The swarm explores the building, finds the fire and extinguishes it by releasing an extinguishing cargo.

        iv. **Outdoor wild fire extinguishing**. In this final scenario, a drone releases a swarm of robots (ground, moving balls) near a wildfire in a forest. The swarm moves to the fire and extinguishes it like in the previous scenario.

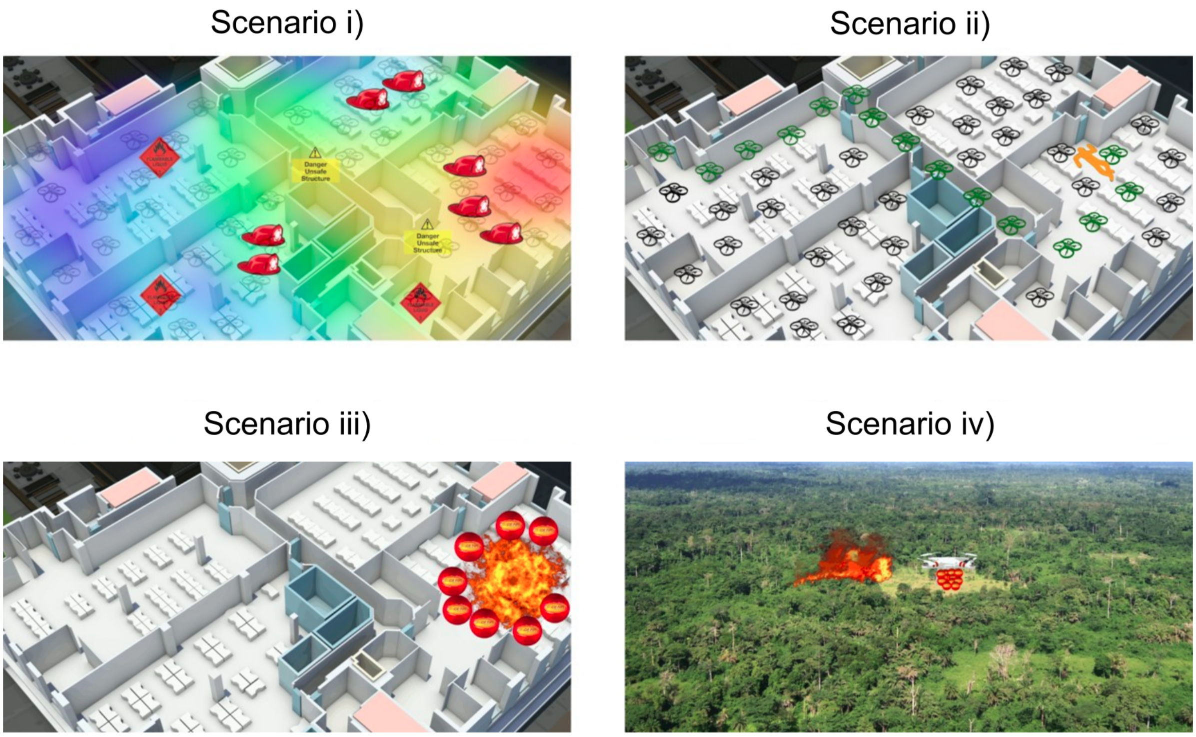


Figure 5.1: Possible scenarios of application shown in the study with fire & rescue services. *i)* The swarm explores the environment to collect information in a building on fire. *ii)* The swarm explores the environment and finds the shortest path between the entrance of a building on fire and the casualties inside the building, or it creates communication links. *iii)* The swarm extinguishes a fire in a building. *iv)* The swarm extinguishes a wildfire in a forest. *Indoor map image modified from Valzania and WRLD3D [2019]. Forest image belongs to public domain (CC0).*

Figure 5.2: Robotic platform designed by Georgios Tzoumas for his masters dissertation in a test scenario at Avonmouth Fire & Rescue station [Tzoumas, 2019].

3. **Discussion of scenarios.** Finally, a group discussion of the previous scenarios, and others suggested by participants, was held between participants and researchers. This part was used to identify the way forward to successfully apply robot swarms to their fields in the future. Participants were verbally asked the following questions:

    a) Can you think of robot swarms being used in other scenarios/for other tasks?

    b) Would you rather use only one robot, a few robots or a very large swarm?

    c) Would you rather use completely autonomous, semi-autonomous or teleoperated robots?

    d) What are the challenges for robot swarms to become a tool in firefighting/rescuing?

    e) What requirements for such a robot swarm would you have?

In addition to the previous parts, a paper-based questionnaire was handed out to participants before the start of the session and at the end of it. The same questionnaire and format was used as pre- and post-questionnaire to quantitatively measure their attitudes towards robots in firefighting and rescuing before and after the focus group, respectively. This way I could measure the impact that the mutual shaping sessions had on participants. A Bhapkar test [Bhapkar, 1966; Agresti, 2003] was used to statistically compare answers from both sets of questionnaires. The choice of this statistical method was motivated by questionnaires with some answers not given, as this statistical test compares nominal data (i.e. unsorted categories). Lack of answers were treated as NA, and added to the NA category. This allowed us to also quantify the change within this category, e.g. participants not answering a particular question in the pre-questionnaire but answering it in the post-questionnaire, or vice versa. Participants were given as much time as they needed to answer the following questions:

1. **In your opinion, how useful could robots as a firefighting/rescuing tool be in the future?** (single choice): Not at all useful, slightly useful, moderately useful, very useful, extremely useful.

2. **In which firefighting/rescuing tasks would robots be most useful?** (multiple choice): Risk/incident assessment, mapping the environment, clearing the way, extinguishing fire, locating victims, rescuing victims, communication links, other (please specify).

3. **How likely would you be to accept help from robots in your job?** (single choice): Not at all likely, slightly likely, moderately likely, very likely, extremely likely.

4. **In your opinion, how many robots would be most useful for firefighting/rescuing?** (single choice): None, only one, a few (no more than a few dozens at the most), many (hundreds, thousands or more).

5. **Helper robots for firefighting/rescuing would be most useful in what mode of operation?** (single choice): When fully controlled by human experts, when semi-autonomous (responsive to human experts' instructions), when fully autonomous performing a pre-defined task (for example, extinguishing fire).

6. **When do you think fire brigades should be included in the research and development process of helper robots for firefighting/rescuing?** (single choice): Not necessary ever (fire brigades only as consumers of the product in the market), from the testing stage (after a prototype has been developed), from the very beginning of research and development (the design stages).

A total of 23 participants from 3 different fire & rescue services were recruited, with experience ranging from 1 to 20 years of service, as they verbally stated. Participant recruitment was done via email, word of mouth and on-site visits to fire & rescue services in the UK and Spain. Participants were given an information sheet with a full description of the study and the focus group. They were also asked to sign a consent form to participate in the study and accept audio recording of the session, complying with university ethics regulations for experiments with human participants. Ethical approval was given by the University of the West of England. No demographic data (e.g. age, gender) were collected to speed up the ethics process. The focus of the study was on the answers provided by the participants, instead of analysing where those answers came from (which would require demographics).

Three focus groups were held, one per service. The first focus group consisted of 6 participants from a UK fire & rescue service. There were participants working in the risk intelligence unit, IT, group management, media communication, operational effectiveness in instant ground, technology management and drone piloting. This focus group was held at the Bristol Robotics Laboratory. In the second focus group, 4 firefighters from a fire station belonging to another UK fire & rescue service came to participate. This focus group was also held at the Bristol Robotics Laboratory. Finally, a third focus group was organised at a Spanish fire station with the participation of 13 firefighters. These are summarised in table 5.1.

### 5.2.2 Results

Below I combine the results from the three focus groups held with fire & rescue services to summarise their current processes, challenges and attitudes towards using robots in fire and rescue based on their comments during the focus groups and their responses to the pre- and post-questionnaires.

#### 5.2.2.1 The art of the profession

Nowadays, firefighters are in charge of many different tasks, not only firefighting. Apart from fires, they go to vehicle collisions, major transport incidents and hazmat incidents. They also do urban

Table 5.1: Summary of the three focus groups with participants from fire brigades.

| Focus group | Number of participants | Type of participants | Location |
|---|---|---|---|
| 1 | 6 | From the research and development department | Bristol, UK |
| 2 | 4 | Firefighters | Bristol, UK |
| 3 | 13 | Firefighters | Murcia, Spain |

search and rescue (e.g. when a building collapses), mine rescue, water rescue, animal rescue and community-based roles to educate the public. When facing incidents, the first things they do are related to gathering as much information as possible for their risk assessment decision-making processes. Before handling the incident, firefighters perform quick checks to guarantee their safety first, e.g. they assess that the structure is safe to operate, or locate access/exit points. After enough information has been collected, firefighters start actions, i.e. firefighting or rescuing, until the incident is completely handled. Then, a fire investigation to discover the cause of the incident might take place. When participants were asked during the focus groups about the current tools they use for firefighting and rescuing, they stated that all tools they use are not automated, but require human operation. A summary of the tools that they currently use is given below:

- **Sensors and actuators fitted to buildings.** They said that smoke detectors, heat detectors and water sprinklers assist them before/during firefighting.
- **Thermal imaging cameras.** They are used to create a map of temperatures to look for the source of the fire and casualties. They are particularly useful to predict what is behind areas with difficult access in buildings. Firefighters highlighted how this type of cameras improves their performance:

  > "Thermal image cameras are one of the great tools we've got. So we can actually see in darkness and make our way around."

- **Hydraulics.** They use hydraulic tools to cut through things.
- **Maps in the fire truck.** These maps are used to locate possible risks, water supplies or weather conditions before arriving to the incident.
- **Radio-frequency identification (RFID).** Used for tracking of firefighters.
- **Colcut$^{TM}$ cobra.** A system that uses high-pressure water to pierce through walls and fog when they cannot access the room next door.
- **Teleoperated ground robots (QinetiQ$^{TM}$).** Sometimes they use them to gather information in hazmat incidents:

> "It's got several cameras and a small water jet for testing temperatures rather than actually extinguishing anything. We used to use them with some level of success. "

- **Drones.** Pilots mostly use them to gather information about incidents to make an assessment of scenarios. They have also used them to track people who have gone missing.
- **Air fans.** Used for tactical ventilation, which means creating positive pressure in a building to push smoke out.

#### 5.2.2.2 Their challenges

Participants highlighted their main challenges are related to obtaining enough, accurate and quick information about the incident so that they can feed it into their decision-making processes. In fact, they mentioned they are quite quick in dealing with fires. The challenge for them is to find the location of those fires, and casualties to rescue. They said this is a challenge because many times the information they get is not accurate (e.g. wrong address, wrong type of fire, unreported casualties):

> "In a lot of cases even the information you get [...] is not always 100 percent accurate. The address could be wrong or the actual type of fire could be wrong. It will come in as a hedge on fire and you get there to find a fire in a building. Your site has no persons trapped, there's no persons involved in anything at that point, and you get there and you find that there are. There's always a variable. You have minimal information."

#### 5.2.2.3 Opinions on usefulness of robots for fire and rescue

Participants could see value in using robots for fire and rescue, as shown in the results of question 1 ("In your opinion, how useful could robots as a firefighting/rescuing tool be in the future?") in figure 5.3. In fact, 20 out of 23 participants ticked *very useful* or *extremely useful* in the post-questionnaire. There was a slight shift from *very useful* to *extremely useful* from the pre-questionnaire to the post-questionnaire, meaning that participants' attitudes were already positive before the sessions. Indeed, the Bhapkar test failed to reject the hypothesis that there was a significant change after the session. However, participants did not think robots should be used for all tasks. Results from question 2 ("In which firefighting/rescuing tasks would robots be most useful?") in figure 5.4 show that information-gathering tasks (locating victims, risk/incident assessment, mapping the environment, communication links) were the ones that participants preferred—they were ticked by over half of the participants. Action-based tasks (clearing the way, extinguishing fire, rescuing victims) were ticked less often, by much less than half of the participants. It is worth highlighting that all tasks but *extinguishing fire* were ticked by the same or more participants in the post-questionnaire, compared to the pre-questionnaire. Hence,

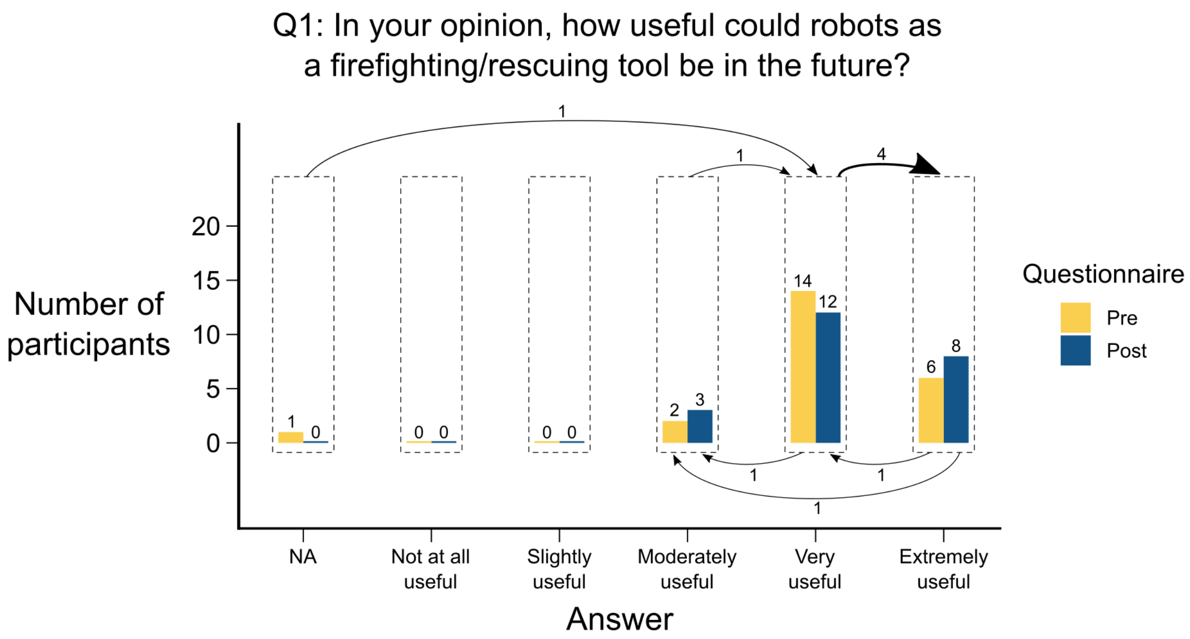


Figure 5.3: Bar chart of attitudes towards usefulness of robots (question 1) in the pre-questionnaire (yellow, left-hand side columns) and post-questionnaire (blue, right-hand side columns). Arrows show the change in answer from the pre to the post-questionnaire.

participants could see more value in using robot swarms after the session, but thought that extinguishing a fire was too complex to be done by robots. Their preference for information-gathering tasks was also highlighted during the focus groups. Participants said that they would prefer robots doing simple tasks, such as going inside a house, mapping it and coming back to them with information; locating casualties; or sending them to gather information before they get to the incident or searching large areas (e.g. ships).

Participants also highlighted the benefit of using robots to create communication links among firefighters (to coordinate their operations) and between firefighters and casualties (to send them reassuring messages). Indeed, one participant mentioned that their research team was looking specifically at what technology they could deploy into a building to have communication across the whole building. Also, they said that there is poor radio communication in many areas where they go, and they would benefit from deploying relays to establish communications in those areas.

Apart from the tasks listed in question 2 of the questionnaire, participants had the choice to specify other tasks that they thought robots could do. In the questionnaires, some participants wrote down the following tasks: hazardous environment identification, post fire investigation (imagery), bring emergency kit (water, oxygen, food, etc.), protection of victims, rescuer and habitable zones. During the discussion, even more examples of tasks were raised, such as:

- **Real-time information.** They said it would be useful to have a swarm of robots deployed across the area of the incident to send constant updates.

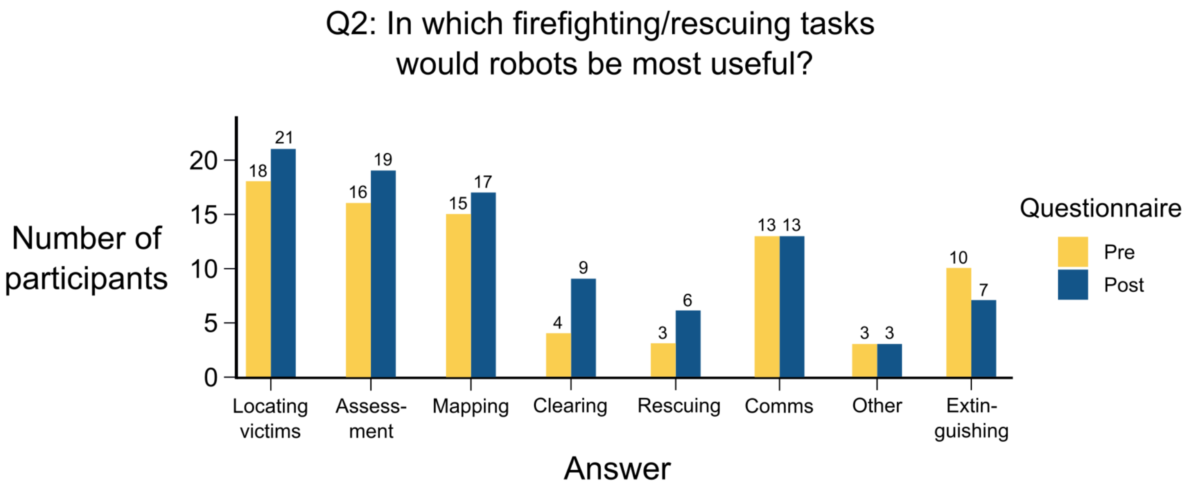


Figure 5.4: Bar chart of attitudes towards applications (question 2) in the pre-questionnaire (yellow, left-hand side columns) and post-questionnaire (blue, right-hand side columns). Order is the categories that increased answers in the post-questionnaire, then the ones that remained the same, and finally the one that decreased answers.

- **Dangerous or repetitive tasks.** They mentioned they would rather have robots where a human being would be in danger, e.g. hazmat environments. Also, some firefighters mentioned they would like to have robots for repetitive tasks, especially to prevent injuries in firefighters.

- **Finding exit routes.** Participants highlighted their difficulties when dealing with heat stress, because they sometimes get confused/lost inside fires. For that, they thought that having robots finding/lighting up the exit route for them would be particularly beneficial:

  > "A building could be like a maze that we're not familiar with [...] You want something that could light up [...] the floor glow [...] something that could glow in the dark. "

- **Tactical ventilation.** Participants gave the example of a swarm of drones using their propellers to perform tactical ventilation to push smoke out of the building.

- **Accessing inaccessible places for firefighters.** Participants said that robots attacking fire in high buildings, where their ladders cannot reach, could be a positive application. They also pictured robots rescuing people from cliffs or water, which sometimes are inaccessible to them.

#### 5.2.2.4 Opinions on acceptance of robots for fire and rescue

Participants answered positively to question 3 of the questionnaire ("How likely would you be to accept help from robots in your job?"). All participants but one ticked *very likely* or *extremely*

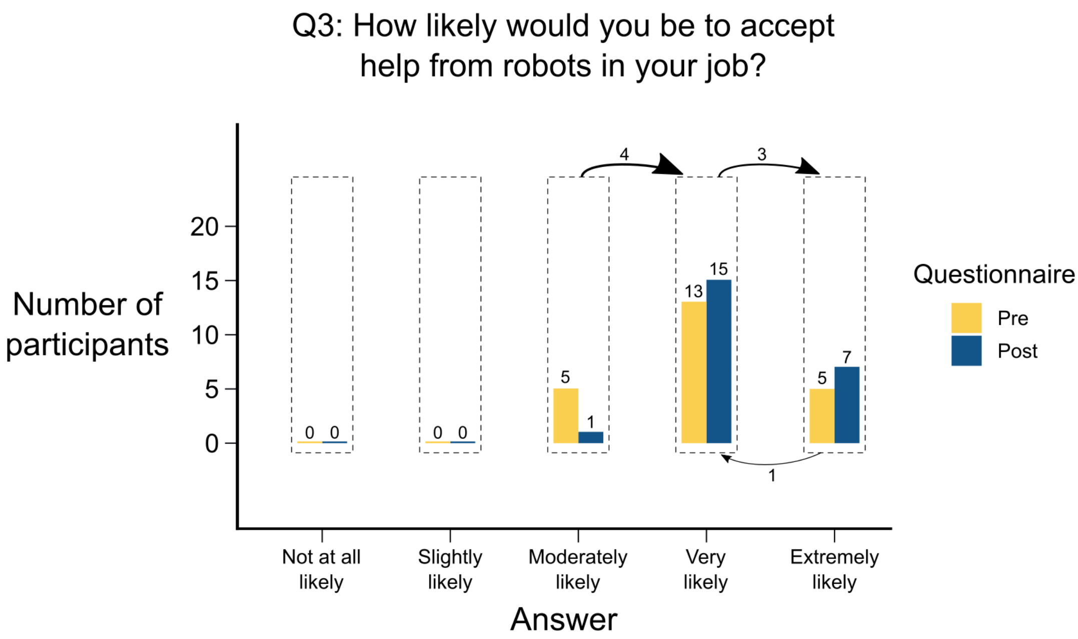


Figure 5.5: Bar chart of attitudes towards acceptance (question 3) in the pre-questionnaire (yellow, left-hand side columns) and post-questionnaire (blue, right-hand side columns). Arrows show the change in answer from the pre-questionnaire to the post-questionnaire.

*likely* in the post-questionnaire, and there was no answer below *moderately likely*, as can be seen in figure 5.5. As in question 1, there was also a slight shift towards more positive answers with respect to the pre-questionnaire, but participants were very positive before the session. The Bhapkar test suggested that there might be a significant change in answers ($\chi^2(2, N = 23) = 6.39, p_{Bhapkar} = 4.1 \times 10^{-2}$). However, a post-hoc McNemar test failed to find the categories with significant change, most likely due to lack of data.

During the focus group, participants pointed out that they do not fear robots becoming a replacement for firefighters. Instead, they see them as a tool that could assist them and enhance/complement their operations:

> “None of us are negative. We all would like it to happen. Yeah it’s just better to have an extra pair of eyes and another person. You just add it to what you’re doing visually anyway. Bring it all together, I can certainly see it being really useful for giving us more information.”

When thinking about acceptance from citizens being rescued by robots (or with the help of robots), participants felt that citizens should be educated. They should know what to expect if robots are used for firefighting and rescuing in the future.

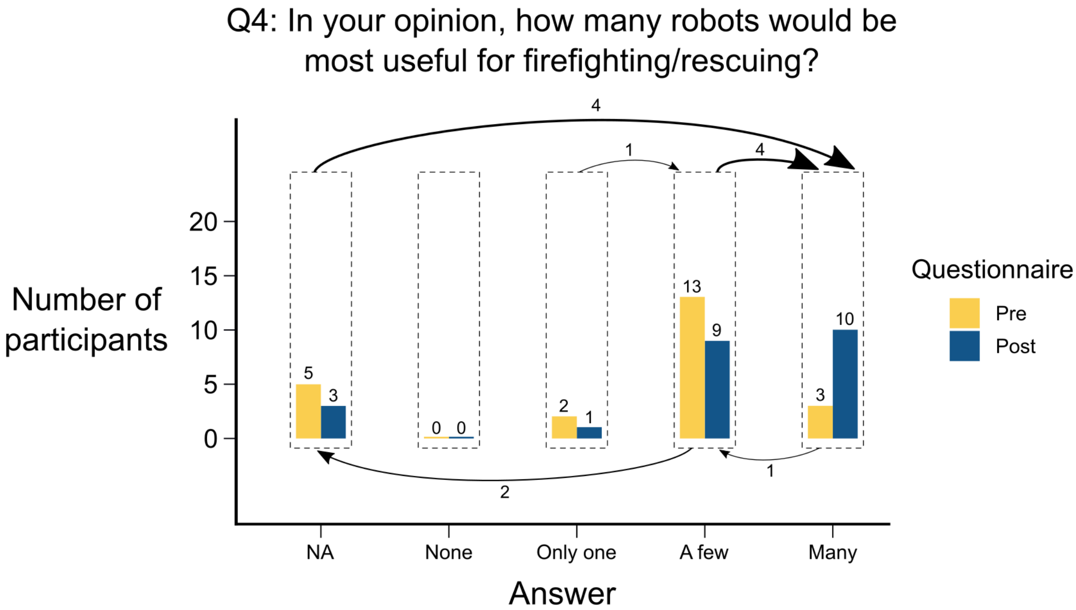


Figure 5.6: Bar chart of attitudes towards number of robots (question 4) in the pre-questionnaire (yellow, left-hand side columns) and post-questionnaire (blue, right-hand side columns). Arrows show the change in answer from the pre-questionnaire to the post-questionnaire.

#### 5.2.2.5 Opinions on robots swarms for fire and rescue

After the session, participants could see how using a large swarm of robots may be the most advantageous option. In question 4 ("In your opinion, how many robots would be most useful for firefighting/rescuing?"), using *many* robots came out as the preferred choice by 10 participants, over *a few* (ticked by 9 participants) and *only one* (ticked by only one participant) in the post-questionnaire. It is worth mentioning that 2 participants did not answer this question, and another one ticked both *a few* and *many*, which was not allowed. Thus, it was not included in the graph of figure 5.6. Remarkably, using *many* robots was ticked by only 3 participants in the pre-questionnaire. Therefore, participants did see the advantages of using a swarm of robots after the sessions. The Bhapkar test suggested that there might be a significant change in answers ($\chi^2(2, N = 23) = 9.53, p_{Bhapkar} = 2.3 \times 10^{-2}$). However, a post-hoc McNemar test failed to find the categories with significant change, most likely due to lack of data too.

During the group discussions, participants understood the base principles of swarm robotics, and highlighted their benefits for fire and rescue. In particular, they said that redundancy is one of the key benefits. Most participants preferred to use a robot swarm even if robots could become obstacles (but left this as a requirement for the future). Also, most participants commented that having a large number of robots would be very useful to quickly search an area and gather as much information in the least amount of time as possible:

"The whole idea around swarm is you got some redundancy built in. [...] And some of

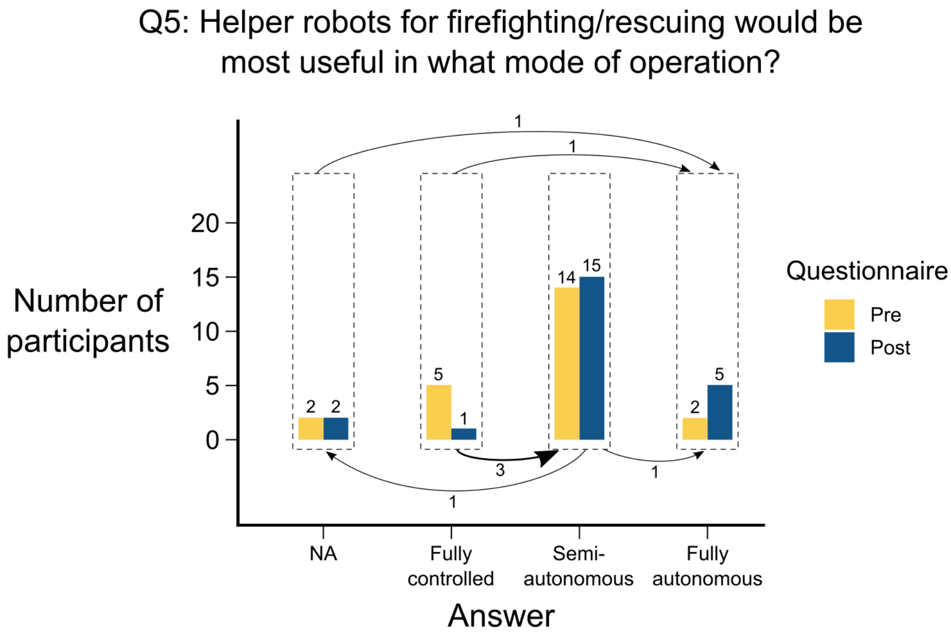


Figure 5.7: Bar chart of attitudes towards mode of operation (question 5) in the pre-questionnaire (yellow, left-hand side columns) and post-questionnaire (blue, right-hand side columns). Arrows show the change in answer from the pre-questionnaire to the post-questionnaire.

> the things we talked about is about location of casualties when it's dark. So deploying small agile devices that can search the rooms at the same time so that firefighters go in and then at least it's a beeping sound, 'yes, okay, let's prioritise that room.' [...] I think those sorts of things would be our friends."

#### 5.2.2.6 Opinions on autonomy

Their preferred mode of operation for robots is semi-autonomy (15 responses in the post-questionnaire), as seen in results for question 5 ("Helper robots for firefighting/rescuing would be most useful in what mode of operation?") in figure 5.7. In fact, this was the participants' preferred mode of operation before they participated in the session, as seen from the results of the pre-questionnaire. Indeed, the Bhapkar test failed to reject the hypothesis that there was a significant change after the session. The session made them mostly abandon the idea of having *fully controlled* robots. It is worth mentioning that answers from 2 participants who ticked *fully autonomous* and *semi-autonomous*, and *semi-autonomous* and *fully controlled* were not taken into account. The directive stated multiple answers were not allowed, hence these answers were discarded.

From the group discussion, I understood that participants did not like the idea of robots taking autonomous decisions. They would trust robots carrying out information-gathering tasks or simple actions rather than stepping in the firefighters' decision-making process. Basically, participants feared that the robot system could cause more harm than benefits (e.g. knock-on effects) because there are many variables during fire and rescue, and lives at risk. They gave the example of robots opening up a window and changing the dynamics of the fire due to a change in air flow and the addition of oxygen to it.

In their opinion, robots could support their decision-making processes, but should not be in charge of them. From their comments, they would rather have a human in the loop being responsible for the actions taken when handling the incident:

> "If it is autonomous just for firefighting, then I don't think that this is a corporate risk we would accept in this site. You can just imagine the headlines, it can help you and save you a thousand times. But one time it doesn't work properly and we lost a building through fire. Or loss of life even worse. Imagine the headlines: 'Firefighters sit outside and do nothing while robots sacrifice and get it wrong'. That's a risk that, until the idea is developed and understood more widely, probably we would not accept."

#### 5.2.2.7 Opinions on involvement in the research and development process

The final question was related to when fire & rescue services should be included in the research & development process ("When do you think fire brigades should be included in the research and development process of helper robots for firefighting/rescuing?"). A total of 16 participants answered *from the very beginning*, whereas only 6 participants ticked *from the testing stage* in the post-questionnaire. One participant did not answer this question, so it does not appear in question 6 in figure 5.8. This aspect was not discussed during the focus groups. As seen in the answers to this question in the pre-questionnaire, mostly the same number of participants already thought that fire brigades should be included from the beginning. Their participation in the sessions did not change this opinion. Indeed, the Bhapkar test failed to reject the hypothesis that there was a significant change after the session.

#### 5.2.2.8 Requirements for robots that assist in fire and rescue

This final section summarises all the key requirements that participants felt robots used in firefighting and rescuing should have for them to trust these systems:

- **Robots should be easy and quick to learn, deploy and maintain.** Participants said that setup time should be kept to a minimum to proceed as soon as possible, as well as the number of checks needed to maintain the robots because they do not have enough time. Cost of training should also be low, according to them.

- **Swarms should not become a physical obstacle for firefighters/casualties.** Firefighters described that fires are usually chaotic, with unpredictable conditions.

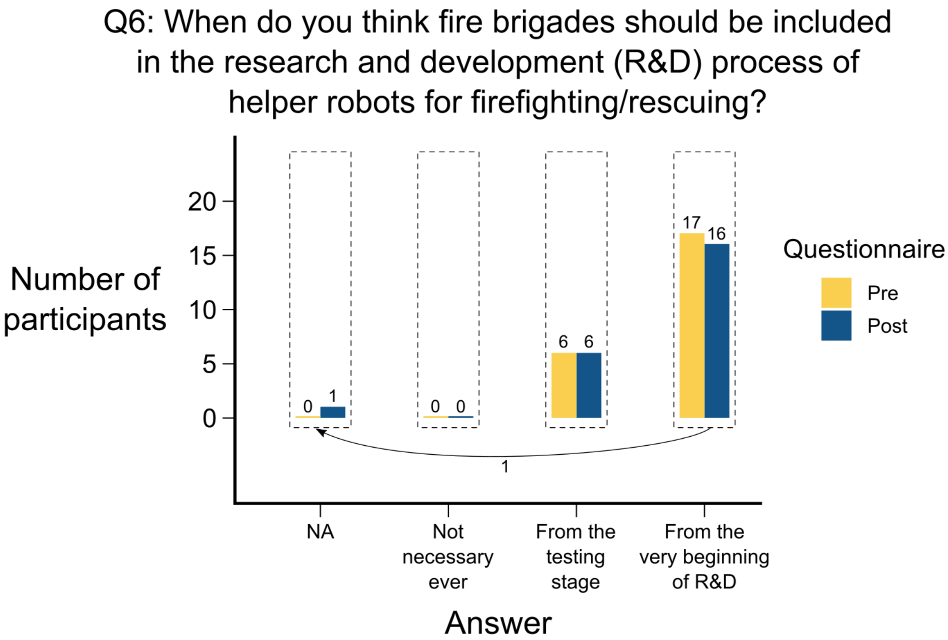


Figure 5.8: Bar chart of attitudes towards involvement (question 6) in the pre-questionnaire (yellow, left-hand side columns) and post-questionnaire (blue, right-hand side columns). Arrows show the change in answer from the pre-questionnaire to the post-questionnaire.

- **Info given from robots should be relevant and not complex.** Due to the amount of information they manage when dealing with an incident, firefighters said that robots should not give all, raw information, but as clear, relevant and digested as possible.

- **Robot swarms should be reliable.** They stressed the importance of guaranteeing that robots work when deployed, and that the information they provide is accurate. This is because they would make decisions based on what robots tell them:

> "The reliability needs to be on there because again, the first time it fails that's it, you've lost the cause in there. [...] Get through those cultural barriers and then you'll find that the actual application implementation of that would be a lot easier."

- **Robot swarms should be accountable.** Participants said that the data gathered from robots must be stored and timestamped. This is important for their internal investigations.

- **Robot swarms should be safe.** Finally, participants said that robots should guarantee firefighters' safety.

### 5.2.3 Discussion

In this section, the main take-home messages to help shape future responsible and successful deployments of robot swarms in the physical realm are summarised from the results described in the previous section.

#### 5.2.3.1 There is opportunity for swarm robotics to assist in fire and rescue

Participants welcomed robots for certain tasks, especially robot swarms. For them, the main advantages are the ones related to robustness via redundancy (no single point of failure) and high performance due to the use of a large number of robots. These properties would be helpful in scenarios that participants felt were most useful, as identified in the focus groups (real-time information gathering, dangerous tasks, communication channels, finding exit routes, testing for hazards/traps, victim location/tracking, tactical ventilation), and from the questionnaire (locating victims, risk/incident assessment, mapping the environment, communication links), as Driewer et al. [2005] also found in their study. In these applications, high speed and large area coverage are common aspects, hence benefiting from a robot swarm collectively performing them in parallel.

#### 5.2.3.2 Identifying the "art of their profession" will inform tasks to be automated

I could see that participants welcomed technology that can assist them with certain tasks, but not all of them. Participants' priority would be on robots that could assist them with information-gathering (e.g. locating casualties, mapping, communication links) or simple actions (clearing the way, lifting heavy things, tactical ventilation) with no autonomous decision-making process in place. This preference can be explained from two different points of view. On the one hand, participants pointed out that finding the fire/casualties and gathering information for their decision-making processes are the main challenges they face. On the other hand, they highlighted their fear that robots making autonomous decisions could cause more harm than good because of unforeseen consequences (many factors are in place during firefighting and rescuing), or lack of understanding of such decisions. Particularly interesting is their preference for not having fire extinguishing robots. Participants felt that there were many aspects involved in firefighting, and that only them, humans, would be capable enough to extinguish fires. This suggests there are certain aspects of their profession that they would not like automated, but done by humans—the art of their profession.

Participants agreed they would rather have robots supporting their decision-making processes as much as possible, but not acting autonomously when it comes to making decisions. Takayama et al. [2008] also found that robots were not preferred for occupations that require evaluation and judgement. Semi-autonomy, meaning that robots can perform some tasks by themselves but always subject to human input (human in the loop), is the preferred mode of operation.

Semi-autonomy was also the preferred mode of operation in the study done by Driewer et al. [2005] and Delmerico et al. [2019]. This suggests that a good human-swarm interaction approach should be pursued.

Robots are often negatively portrayed as machines taking over jobs. The fact that there are some aspects of their profession that participants would like to protect could seem to be related to this, although a direct question about fear of losing their jobs was not asked to participants. Participants broadly welcomed the use of robots in their jobs, and agreed they would be a tool to enhance/assist in their operations rather than a replacement. This is similar to the findings of the survey by Takayama et al. [2008] in which non-expert participants were more likely to prefer robots in a given occupation with people, rather than instead of people. Taking into account that there are barely any robot swarms currently in place for fire & rescue services, the fact that participants welcomed their use for certain tasks shows a high degree of preliminary acceptance. Therefore, when looking at how to best deploy robots in the physical realm, it is important to identify with end users which aspects are/are not desired to be automated to increase user acceptance.

#### 5.2.3.3 There are concerns to tackle to increase acceptance and trust

Participants were mainly positive about robot swarms and the applications in their fields. However, there were caveats in each case, meaning that participants would trust swarm robotics systems under certain conditions. It is then crucial to address these concerns, if a successful implementation in society is sought. In fact, user acceptance and trust have been identified as the major bottleneck when taking robots to real-world applications [Kruijff et al., 2014].

- **Transparency and accountability**

  Participants pointed out that robot swarms should always store all the data they generate or process–timestamped. It is very important for them to understand *what* the swarm is doing, especially in case an investigation is required. In this sense, the swarm must be accountable, i.e. able to be queried and return a human-understandable answer. This is the concept of an ethical black box, described by Winfield and Jirotka [2017] as a mechanism to improve public trust by designing robots with accountability at the core.

- **Reliability and safety**

  For firefighters, another aspect to help build trust in robot swarms is reliability, i.e. the guarantee that if the robot swarm is deployed in a fire & rescue operation, it will work properly. In the scenarios they face, faults cost lives. Hence, all the information that the robot swarm might gather or the actions they perform must be completely accurate. This requires thorough verification and validation of the swarm robotics system before deployment. However, predicting the emergent collective behaviour of robot swarms given the individual rules of each robot, and making sure that it is the only behaviour that the swarm shows is a major challenge [Dixon et al., 2012]. Further research on designing reliable swarms should be prioritised to increase trust, as well as reduce the number of risks arising from the use of swarms [Harbers et al., 2017].

Safety also came out as another requirement for acceptance and trust in the focus group with fire brigades. The robot swarm not becoming a physical obstacle (either for firefighters or casualties) was especially regarded as a crucial feature of the swarm robotics system. As argued above, this has to do with the requirement for robots not being detrimental to their operations. These safety concerns indicate that for swarm technologies to be accepted in the future, relevant safety standards will need to be developed [Beltrame et al., 2018; Bjerknes and Winfield, 2013; Winfield et al., 2004].

- **Ease of training, use and maintenance**

  Most participants agreed that they would trust the robot swarm assisting them at work as long as it was easy to learn about, use and maintain. Time is a crucial aspect for firefighters. Hence, they require a system that can be deployed fairly quickly (ready by the time they arrive to the incident location), not too complex to use (their cognition abilities are harmed when firefighting, for example) and that does not require complex maintenance (always ready to be used). This places the focus on the scalability and adaptability of the robot swarm operations. Essentially, this means that if an action has to be done on the swarm, it should be independent of the number of robots in the swarm or the location of deployment.

  These results are in line with the findings from Yanco et al. [2006], where participants expressed their desire for the system to be easy to use—in fact, the system being difficult to use was the main cause for their test missions failing. Moreover, participants from the study led by Driewer et al. [2005] preferred an easy-to-use system. Authors then suggested having the ability to select different layers of information depending on what the specific user might require. This could indeed improve adaptability of the systems to users.

#### 5.2.3.4 Mutual shaping can facilitate the deployment of robot swarms in the physical realm

The analysis of the responses to the pre- and post-questionnaire was used to understand the role of mutual shaping through focus group discussions in changing their opinions. The following changes in attitudes were noticed:

- **More tasks for robots.** In terms of tasks where robots could be useful, there was an overall increase in all tasks after the session (except extinguishing). This tells us that the session made them see how robots could be used for more tasks than they previously had thought.

- **Acceptance of robots increased.** There was over 20% increase to the *very likely* or *extremely likely* responses to the question related to acceptance of assistance from a robot. A total of 18 participants ticked any of those in the pre-questionnaire, whereas 22 participants ticked any of them in the post-questionnaire.

- **A large swarm of robots is preferred.** When firefighters were asked about the number of robots they would rather have assisting them, *a few* was the most selected answer (13 participants), whereas *many* was chosen by only 3 participants. After the session, 9 participants ticked *a few* and 10 participants ticked *many*.

Mutual shaping has been shown to be a successful way to engage in a two-way conversation with potential users and incorporate societal choices into the research and development process. If robot swarms are to be used in real-world applications, it is important to listen to all the parties who will be affected by it in the future. Almost three quarters of the firefighters said that they would like to be involved in the research and development process from the very beginning, in both questionnaires.

## 5.3 *Swarm Escape!*: An escape room experience to engage the public in swarm robotics

This study consisted of two activities: a swarm-robotics-themed escape room named *Swarm Escape!*, and a discussion session named *Robot Swarms in our Cities*, both held at the Festival of the Future City 2019 in Bristol (United Kingdom). This study was also funded through an EPSRC Impact Acceleration Account award.

The escape room was created as a way to introduce people to the topic of swarm robotics and current research on the topic. Furthermore, it is important to educate society about future uses of swarm robotics (based on the comments from fire brigades expressed during the previous study), as they will be impacted by it, or even become the users of tomorrow. The discussion session allowed participants to think about future uses of robot swarms by themselves, express their opinions about opportunities and concerns around them, and come up with actions that researchers/companies/governments should implement for robot-swarms applications to become a successful reality in the future.

### 5.3.1 Methodology

#### 5.3.1.1 *Swarm Escape!* escape room

To make the escape room a successful experience, I designed it to be:

- **Fun.** The most important aspect was to make the experience fun, as this improves learning [Carrillo-Zapata et al., 2020]. Therefore, the escape room experience had be similar to the one from a commercial escape room (what players would expect). In practical terms, this meant creating different types of puzzles, hiding clues/objects, having a narrative and a goal to "escape", etc.

- **Educational.** I wanted to i) show current research on swarm robotics and a possible application in the future, ii) guide players through the stages of research and development of robot swarms, and iii) show the main properties of swarm robotics. These were achieved by showing videos of swarm robotics experiments, designing the experience so that players had to follow different stages of research and development, and giving cards with a different property of swarm robotics when players solved each puzzle.

- **Not time-constrained, but solvable in 30 to 60 minutes.** As opposed to a commercial escape room, the focus of an educational escape room is on learning. Therefore, using a timer to constrain players to a limited time to solve it (typically one hour) would put the focus on them rushing to solve all puzzles, discouraging reflexivity, hence, learning. However, puzzles should be adapted to what users would expect from a commercial escape room. Therefore, difficulty of puzzles had be carefully tailored to allow players to solve it in less that one hour, but no less than 30 minutes to still make them feel challenged. Trialling the escape room with different players and a few times before launching it helped adapting the puzzles.

- **Inclusive.** As shown by Nicholson [2016], escape rooms are balanced in terms of gender, with most groups being mixed. Apart from gender inclusion, I designed *Swarm Escape!* to allow players from any age or background in (swarm) robotics or STEM to play, just like commercial escape rooms. Therefore, no previous knowledge was required to play, and puzzles were designed to be solved by traditional escape-room-methods, e.g. finding objects to assemble in order to reveal a code, placing objects in the right spot, etc. In addition, access to the escape room was suitable for people with reduced mobility (when mounted outdoors), and colour clues were adapted to colour blindness.

- **Portable.** As opposed to commercial escape rooms that are fixed in a place, *Swarm Escape!* was designed to be portable. Thanks to this feature, it can be set up practically anywhere, either outdoors or indoors. The advantage is an increase in reachability, as it could be set up in public spaces (e.g. parks, squares), festivals (e.g. science festivals), inside buildings (e.g. business, universities), either nationally or internationally. This was obtained by designing easy-to-set-up puzzles independent from a physical space. To mount it outdoors, a portable gazebo was used.

The escape room was built around a fictional scenario, with players becoming swarm robotics researchers, and tasked with "designing" and "programming" a robot swarm to help emergency services mitigate a pollution crisis. Research done in §5.2 showed that users are open to the use

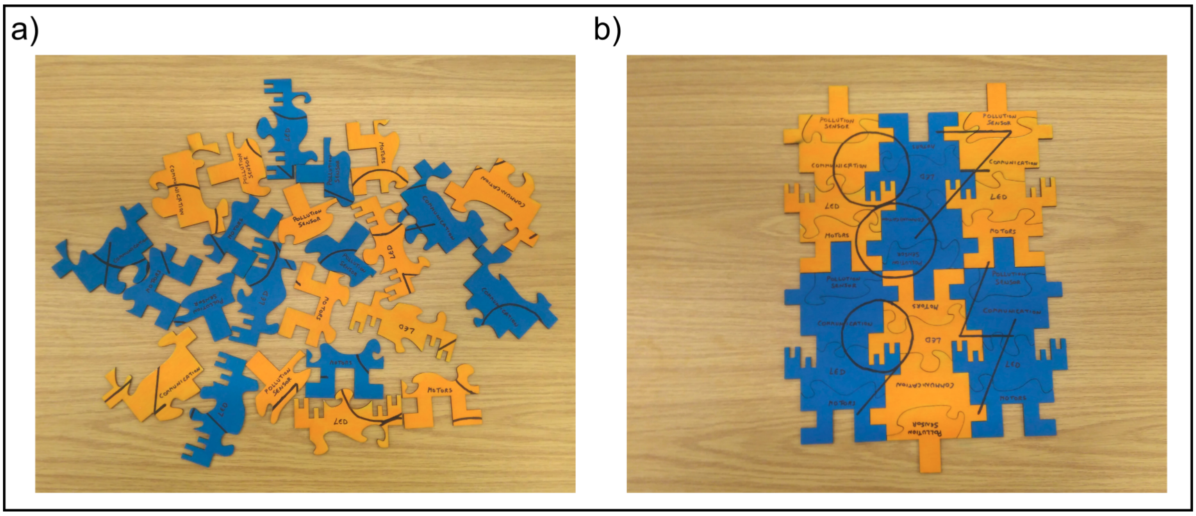


Figure 5.9: *a) Swarm Escape!* puzzle 1 before solving. *b) Swarm Escape!* puzzle 1 solution.

of swarms to assist in fire & rescue operations, hence motivating the escape room plot. Here is the summarised version of the plot that was used for advertising in the project website[1]:

*A mysterious pollution cloud has arrived in the city.*
*The sky has turned black.*
*Air is unbreathable.*
*Emergency services are evacuating the population.*
*You and your emergency technology special unit are the last hope to combat the pollution.*
*Will you be able to do it before the city is devastated?*

There are three specific goals that players have to achieve to successfully finish the game: 1) build a robot swarm, 2) program the robot swarm to find the most dangerous pollution clouds, and 3) program the robot swarm to find the fastest and safest evacuation route. These goals are directly linked to the three main puzzles of the game, i.e. solving a puzzle means achieving its particular goal. These main puzzles have to be solved sequentially to guide players through a meaningful research and development process, although some clues are available at stages different to when they are really needed in order to add non-linearity to the game, hence, adapting difficulty and playfulness. A description of the main puzzles is given below:

- **Puzzle 1: Building a robot swarm.** This puzzle consists of finding hidden jigsaw puzzle pieces, and assembling them to reveal a hidden number code (see figure 5.9). When the jigsaw puzzle is solved, six robots appear assembled as a tessellation, i.e. the same robot shape (the tile) is used for the six robots. In turn, each robot is formed by four jigsaw puzzle pieces representing the basic components of a simple swarm robot: pollution sensor,

[1] `http://www.swarmescape.org/`

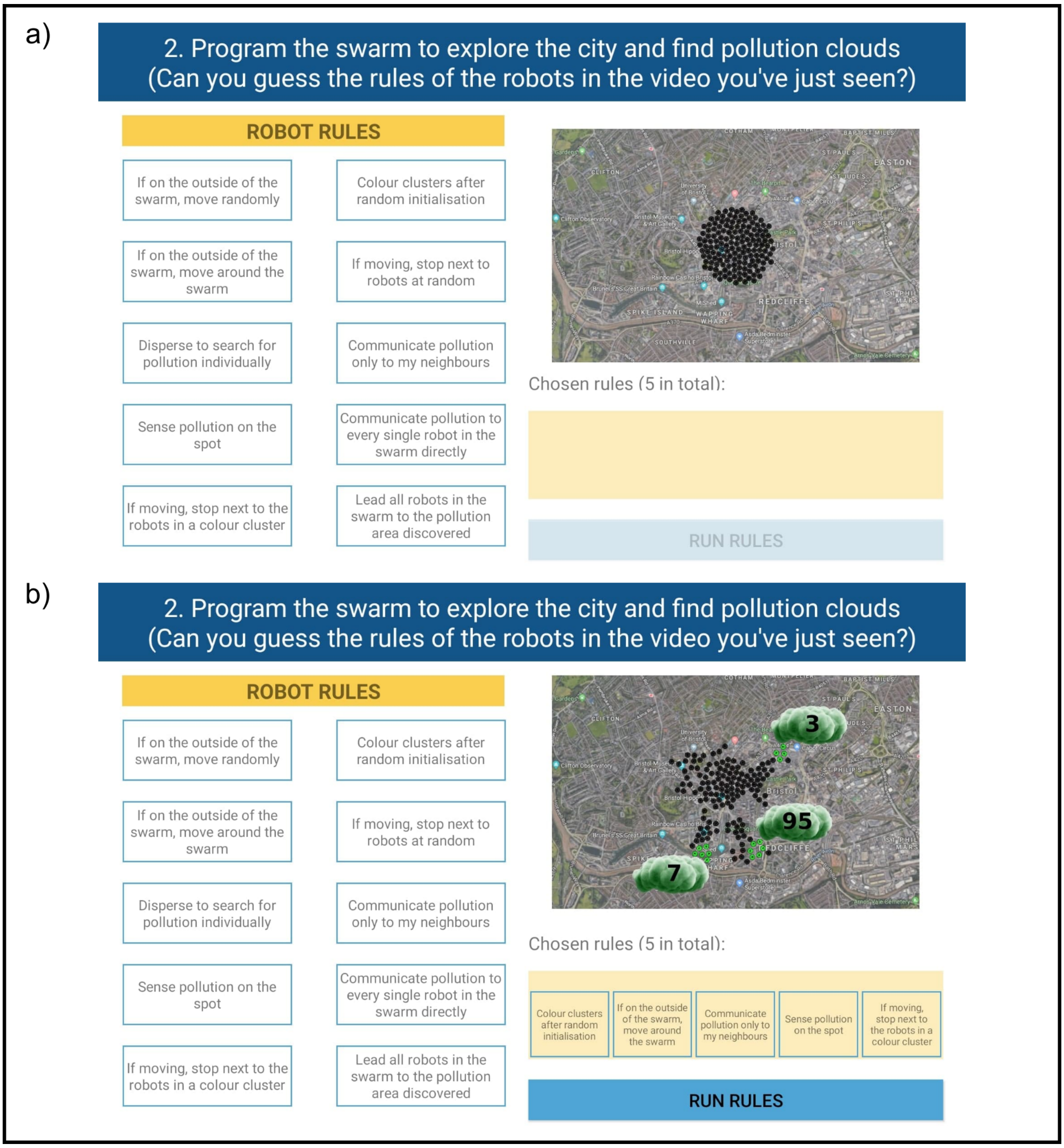


Figure 5.10: *a) Swarm Escape!* puzzle 2 before solving. *b) Swarm Escape!* puzzle 2 solution.

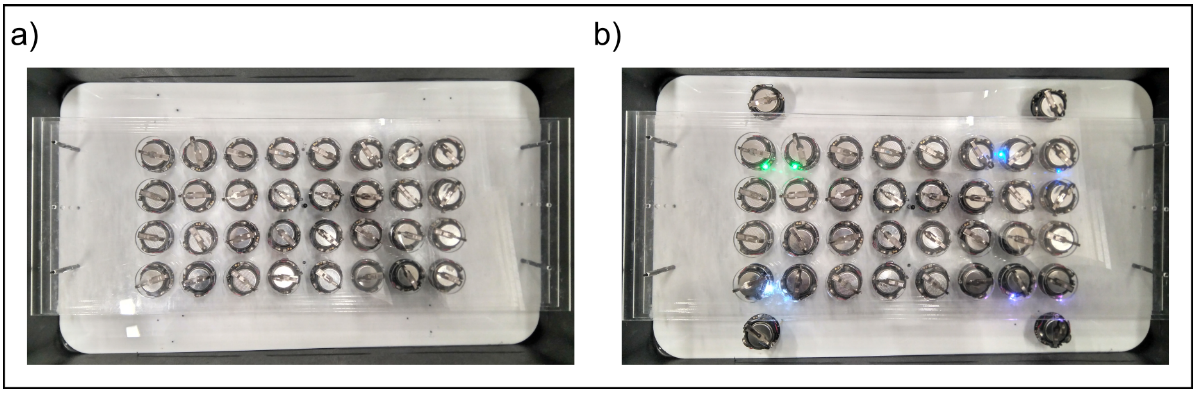


Figure 5.11: *a) Swarm Escape!* puzzle 3 before solving. *b) Swarm Escape!* puzzle 3 solution.

communication, LEDs (to visualise the state of the robot), and motors. The fact that the jigsaw puzzle is a tessellation–meaning that the same robot shape can fill a mathematical plane with no overlaps or gaps–is used as a way to convey the message that robots in a swarm must work together. When this puzzle is solved by the players, they obtain a card with the property of robot swarms "large number of simple robots".

- **Puzzle 2: Programming the robot swarm to find the most dangerous pollution clouds.** To solve this puzzle, players first have to watch a video showing an experiment in simulation that corresponds to the research done in chapter 4. Then, on a tablet, they have to select the five rules the robots follow to perform their swarm behaviour, given a choice of 10 different rules (see figure 5.10). It is worth highlighting that brute-force is discouraged and minimised by having the tablet lock itself during two minutes each time players choose incorrect rules three times in a row. Instead, this promotes discussion among the team, which is a seed for learning. When this puzzle is solved by the players, they obtain two cards with the properties of robot swarms "simple rules" and "emergent collective behaviour".
- **Puzzle 3: Programming the robot swarm to find the fastest and safest evacuation route.** For this puzzle, players have to use real swarm robots. Concretely, they are given a small swarm of 36 Kilobots, and they have to discover how to make them interact with each other to light up different colours representing different exit routes for citizens (see figure 5.11). When this puzzle is solved by the players, they obtain two cards with the properties of robot swarms "local interaction" and "no leader".

A picture of the escape room layout before and after a game in shown in figure 5.12. Figure 5.13 also shows a more detailed diagram with the layout and locations of all props. Figure 5.14 shows a diagram with the logic of the escape room, i.e. how puzzles and props are interconnected and used to solve it. When players complete the three puzzles, they are considered to have successfully completed the game. Finally, players obtain a card with the sentence "robot swarms can be useful" written on it.

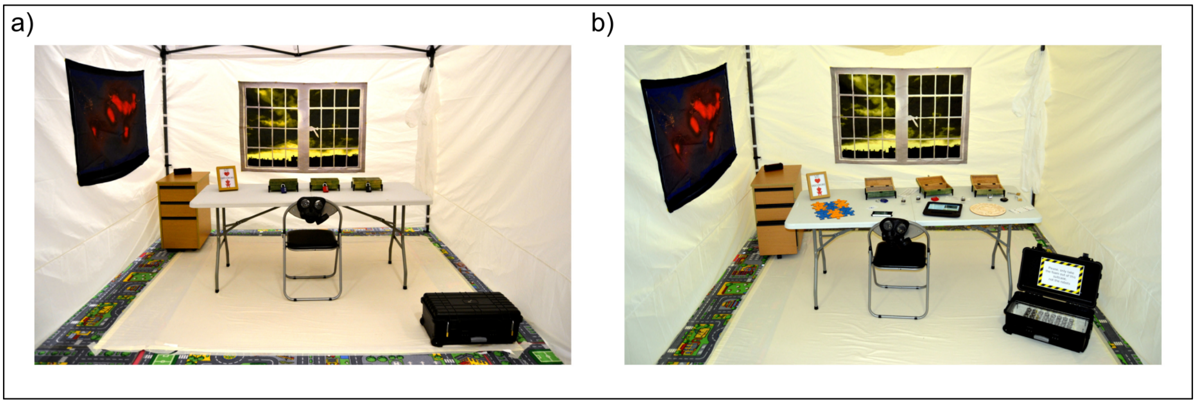


Figure 5.12: *a) Swarm Escape!* escape room setup. *b) Swarm Escape!* layout after a team finished the game.

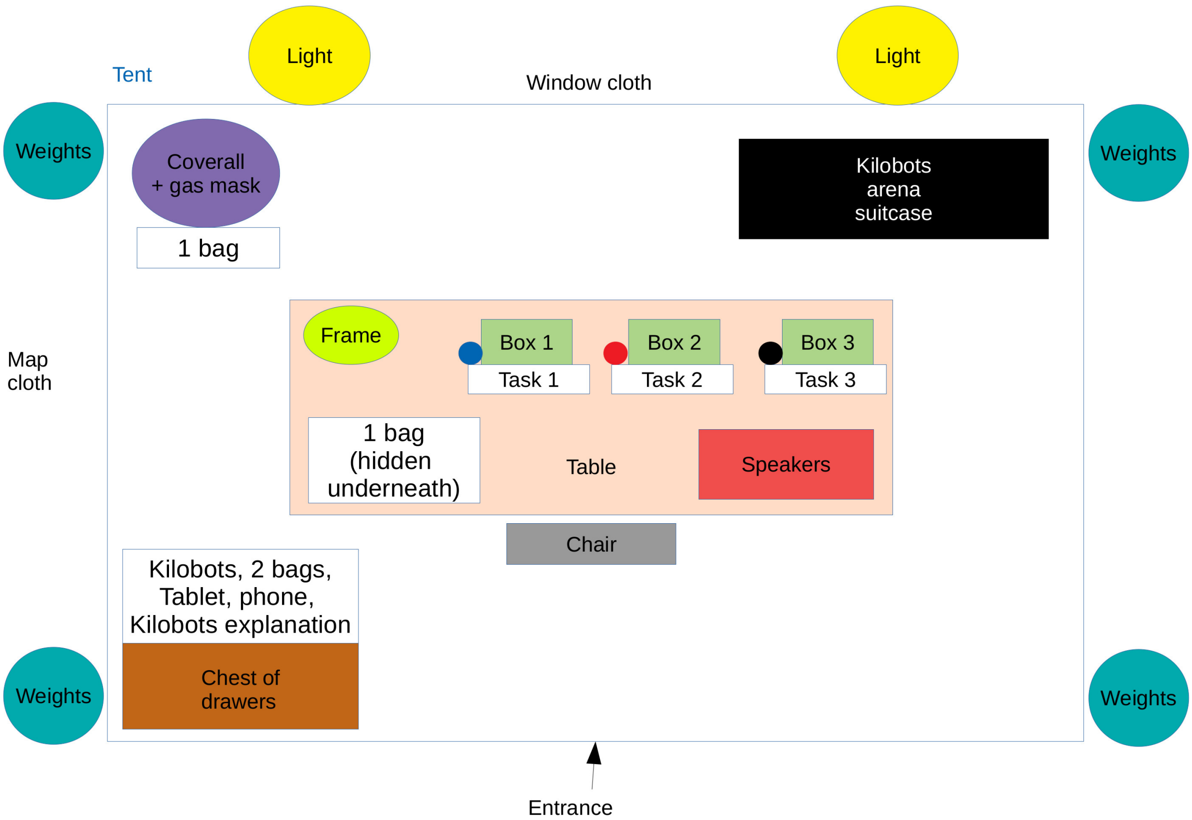


Figure 5.13: *Swarm Escape!* props layout.

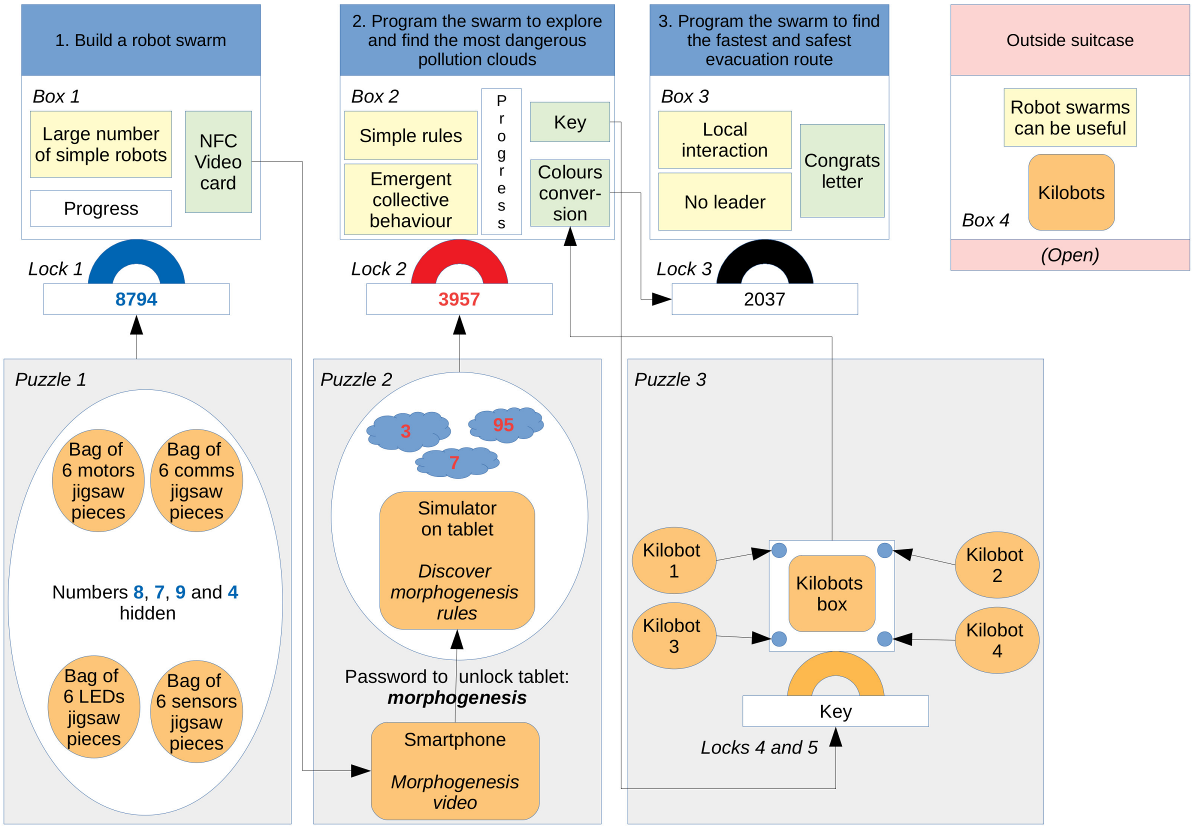


Figure 5.14: Logic of *Swarm Escape!*. At the beginning, four bags with jigsaw puzzle pieces, a password-protected tablet, a mobile phone and four Kilobots are available to players. When the jigsaw puzzle is solved (puzzle 1), players can unlock box number 1 with code *8794*, obtaining and NFC card that triggers the morphogenesis video on the mobile phone. At the end of this video, players are given the password to unlock the tablet (*morphogenesis*). When they guess the correct robot rules on the tablet (puzzle 2), players can unlock box number 2 with code *3957*, obtaining a key to open the big suitcase with Kilobots and a colours conversion table. When players find the right position of the four Kilobots inside the Kilobots suitcase, they can use the colours conversion table to find the code (*2037*) to unlock box number 3. The game is then successfully completed.

To measure impact on learning and attitudes of players, a written pre- and post-questionnaire methodology was used. The same questions were used in both questionnaires to properly measure whether the escape room had an effect on players. The pre-questionnaire was handed out before players entered the escape room. The post-questionnaire was handed out when players completed it. The questions were the following:

1. **Do you know what swarm robotics is?** (single choice): yes, no.

   - If answered *yes*, please, write down what you think swarm robotics is.

2. **How would you feel having a robot swarm assist you at work/home?** (single choice): totally uncomfortable; very uncomfortable; moderately uncomfortable; not sure; moderately

comfortable; very comfortable; totally comfortable.

3. **What do you think about the benefits and risks of using robot swarms in our society?** (single choice): benefits of robot swarms for society outweigh risks; benefits and risks of robot swarms for society are about the same; risks of robot swarms for society outweigh benefits; not sure.

Players were recruited by advertising the event on the festival website, the project website and social media, as well as on-site recruitment where the escape room was set up. A total of 40 people played the escape room. Ethics approval was granted by the Ethics Committee of the Faculty of Engineering from the University of Bristol. No demographic data (e.g. age, gender) were collected to speed up the ethics process. The focus of the study was on the answers provided by the participants, instead of analysing where those answers came from (which would require demographics).

#### 5.3.1.2 *Robot Swarms in our Cities* session

For this session, I invited both players from *Swarm Escape!* and people who had not played it, to discuss robot swarm in real-world applications. Effectively, this was a way to engage in a dialogue with members of the public about swarm robotics and its future implications for society. The session started with a short introduction to swarm robotics. Then, participants were split into groups of 3-5 people each, plus a swarm robotics researcher from the Hauert Lab[2]. This part of the session was structured following the principles of mutual shaping to establish a two-way relationship between researchers and society to incorporate societal choices in the research and development process. Essentially, these groups were small focus groups discussing the following three questions:

1. **What can you imagine robot swarms doing?** Participants were asked to individually write down any applications of robot swarms they could think of in the form of sticky notes. Then, they were invited to stick them to a benefits-versus-risks graph hanging on the wall.

2. **What are the opportunities for/concerns about robot swarms?** In this discussion, participants were asked to express their thoughts about the opportunities of robot swarms as well as their concerns. These were also individually written down in sticky notes.

3. **What should everyone be doing to make robot swarms a reality?** The last question asked participants to think of and write down any actions that should be done by researchers, companies, governments, etc., to have useful robot swarms deployed in the real world.

[2] http://hauertlab.com/


Figure 5.15: Pictures of some players of *Swarm Escape!* after finishing the game. Participants gave signed consent for these pictures to be used publicly.

Participants were recruited by advertising the event on the festival website, the project website and social media. A total of 12 people participated in this session. Ethics approval was granted by the Ethics Committee of the Faculty of Engineering from the University of Bristol.

### 5.3.2 Results

#### 5.3.2.1 *Swarm Escape!* escape room

The escape room attracted people of all ages, from 16-year-old students to people in their fifties and sixties (see figure 5.15). Teams were also highly balanced in terms of gender, which corresponds to Scott Nicholson's findings [Nicholson, 2016]. In addition, on-site recruitment of passers-by allowed to have players with different backgrounds. Even if no timer was used, i.e. there was no maximum time to finish the game (as opposed to commercial escape rooms), the average finishing time was 40 minutes, with 27 minutes achieved by the fastest team, and 57 minutes by the slowest team. I collected written feedback from players after finishing the game, which certified that they had fun playing *Swarm Escape!*. Some examples are quoted below:

> "The experience was both fun and educational! I know how swarm robotics works now! (In principle, at least)"

> "It was fun. Good way of teaching about robot/swarms and their benefits."

> "I think the puzzles seemed really professional and it was a real escape room! As a participant you have the opportunity to learn as you go but it was also good because there wasn't too much information to digest. It was a really fun way of engaging with a tricky STEM topic."

A quantitative analysis of responses to the questionnaires was done to formally measure impact. Question 1 asked players about their knowledge of swarm robotics. Regarding their previous knowledge of swarm robotics before playing *Swarm Escape!* (figure 5.16, pre-questionnaire), 16 players (40%) answered they did not know about swarm robotics, another 16 players (40%) answered they did know about swarm robotics, and 8 players (20%) did not circle any answer, but

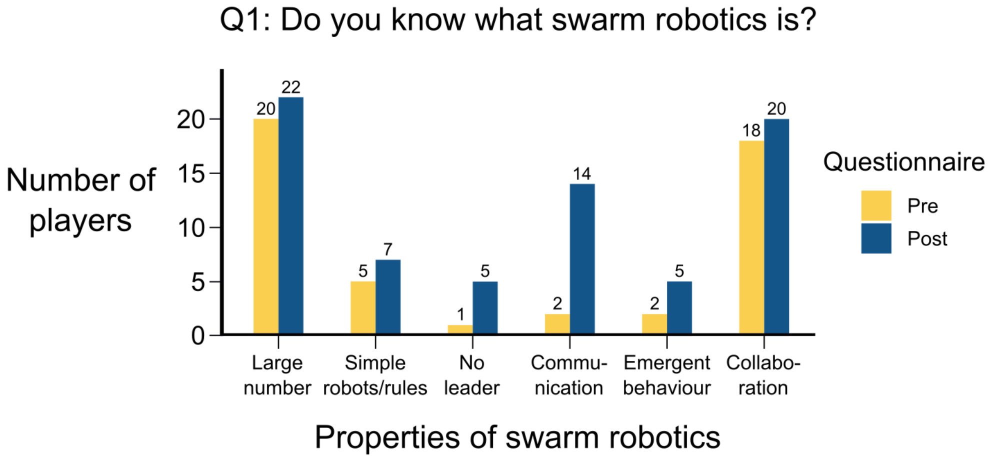


Figure 5.16: Mentions to swarm robotics keywords when defining swarm robotics in question 1 in the pre-questionnaire (yellow, left-hand side columns) and post-questionnaire (blue, right-hand side columns).

actually all of them wrote down a definition in the box. I analysed players' definitions to check whether they had an idea about swarm robotics or not. The criterion was to check whether there was any explicit mention or synonyms to at least one of the main properties of swarm robotics, shown in figure 5.16. This check revealed that 21 players (52.5%) had a previous idea about swarm robotics, but 19 players (48.5%) did not. After players completed the game (figure 5.16, post-questionnaire), 33 players (82.5%) mentioned at least one property of swarm robotics, 6 players (15%) did not, and 1 player (2.5%) ticked *yes* without giving any definition in the box. The most mentioned concepts in both the pre-questionnaire and post-questionnaire were *large number of robots* and *collaboration*. The concept of *communication* received a remarkable increase in mentions in the post-questionnaire, with an increase of 12 players. The rest of the concepts (*simple robots/rules*, *no leader* and *emergent behaviour*) were mentioned by no more than 7 players in either questionnaire, hence perhaps being the most difficult concepts to learn/express when defining swarm robotics. In the future, the properties of swarm robotics will be made more clear (e.g. by giving away keyrings with them printed). Even if not all concepts were mentioned, most players who had not given any written definition of swarm robotics before playing actually wrote something afterwards. Here are some examples:

> "Lots of small robots that can communicate to each other to identify and tackle problems in a city."

> "A group of robots that work together to combat pollution/help society."

> "Small robots communicating with each other to detect hazards."

Question 2 asked players about their attitudes towards having a robot swarm assist them at work or at home. With respect to their previous attitudes towards having a robot swarm assist them at work or at home (figure 5.17, pre-questionnaire), most players answered *not sure* (45%), followed by *moderately comfortable* (32.5%), *moderately uncomfortable* (10%), *very comfortable* (5%) and *totally comfortable* (5%). There were no answers below *moderately uncomfortable*, and 1 player (2.5%) did not answer at all (NA). These results are similar to those in the Special Eurobarometer [Eurobarometer 460, 2017], with slightly over one third of participants choosing *moderately comfortable* or higher. In the post-questionnaire, I could see a shift towards positive attitudes (i.e. *moderately comfortable* and higher), with *moderately comfortable* becoming the most picked choice by 22 players (55%). Arrows in figure 5.17 show how many players changed opinion from the pre-questionnaire to the post-questionnaire. Two players (5%) did not give an answer (NA). Most importantly, the *not sure* choice went down from 18 players to only 2. From these, most players went to the *moderately comfortable* choice. Indeed, a Bhapkar test with a McNemar post-hoc test revealed that this change from the pre-questionnaire to the post-questionnaire was significant ($\chi^2(2, N = 40) = 34.3, p_{Bhapkar} = 2.08 \times 10^{-6}, p_{McNemar} = 3.8 \times 10^{-4}$). This suggests that the escape room made unsure players form a positive attitude towards robot swarms. Although it seems that the escape room helped shift players' opinion towards a positive one regardless of their initial opinion, there was not enough data to confirm that the other answers changed significantly. More runs of the escape room would be necessary to check this.

Question 3 asked players about their attitudes towards benefits and risks of robot swarms. Regarding players' preliminary attitudes towards benefits and risks of robot swarms (figure 5.18, pre-questionnaire), most of them also answered *not sure* (52.5%), followed by *benefits equal risks* (25%) and *benefits overweigh risks* (20%). One player (2.5%) did not answer this question (NA) In the post-questionnaire, the *not sure* choice suffered a reduction of 10 players, while the *benefits overweigh risks* category was chosen by 11 players more, becoming the most picked answer among players (47.5%). A Bhapkar test with McNemar post-hoc test revealed that changes in opinion from the pre-questionnaire to the post-questionnaire for the *not sure* and *benefits overweigh risks* answers were statistically significant ($\chi^2(2, N = 40) = 17.2, p_{Bhapkar} = 6.3 \times 10^{-4}, p_{McNemar} = 3.6 \times 10^{-3}$). This also suggests that the escape room had a positive effect on making unsure players form a positive opinion towards the benefits of robot swarms over their risks. This positive shift could perhaps be the result of having a beneficial application of robot swarms (i.e. search and rescue) as the main topic of the game. However, slightly over one quarter of players were still undecided after they finished. In future iterations, benefits and risks will be explored more during the game.

#### 5.3.2.2 *Robot Swarms in our Cities* session

After being introduced to swarm robotics, participants came up with around 40 different uses of robot swarms in 15 minutes, which they placed in the graph shown in figure 5.19. Some applica-

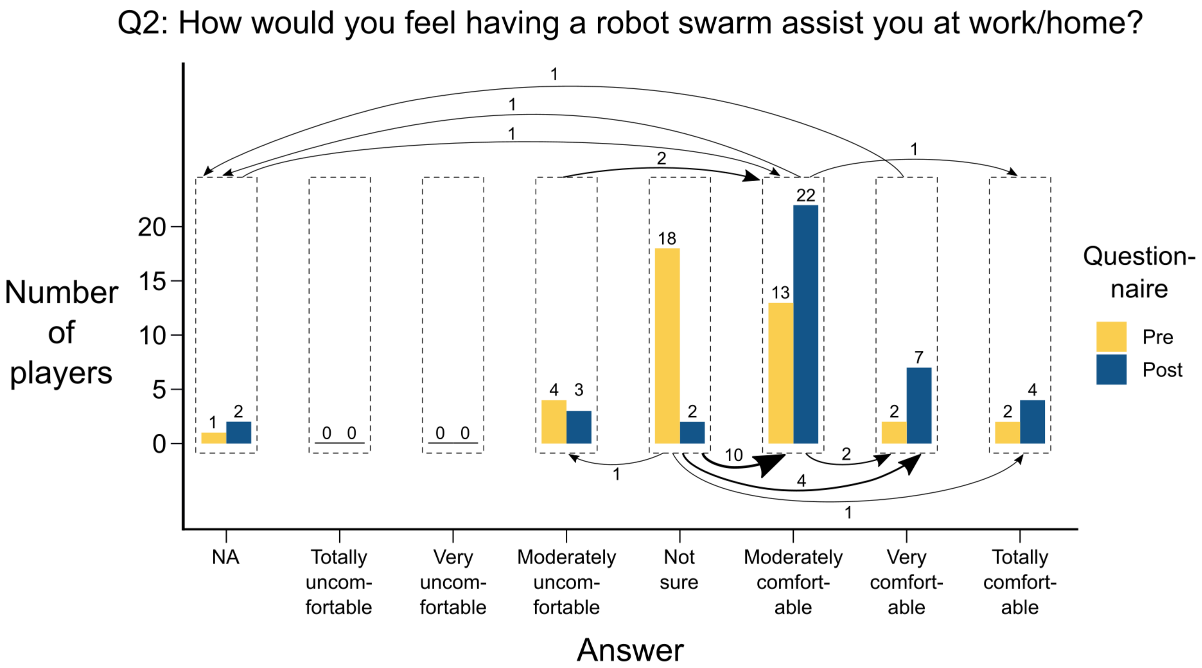


Figure 5.17: Bar chart of attitudes towards having robot swarms assisting players (question 2) in the pre-questionnaire (yellow, left-hand side columns) and post-questionnaire (blue, right-hand side columns). Arrows show the change in answer from the pre-questionnaire to the post-questionnaire.

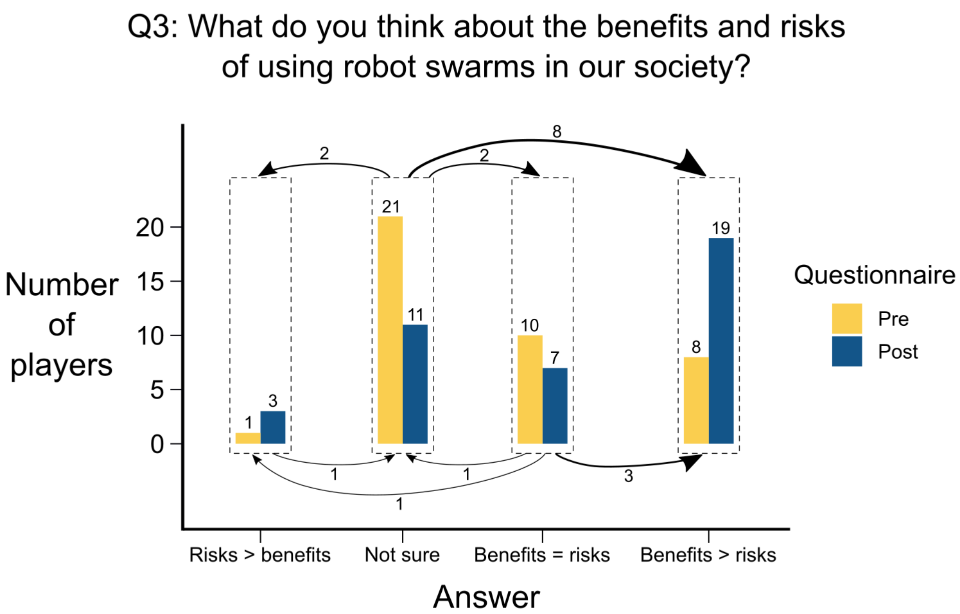


Figure 5.18: Bar char of attitudes towards benefits and risks of robot swarms (question 3) in the pre-questionnaire (yellow, left-hand side columns) and post-questionnaire (blue, right-hand side columns). Arrows show the change in answer from the pre-questionnaire to the post-questionnaire.

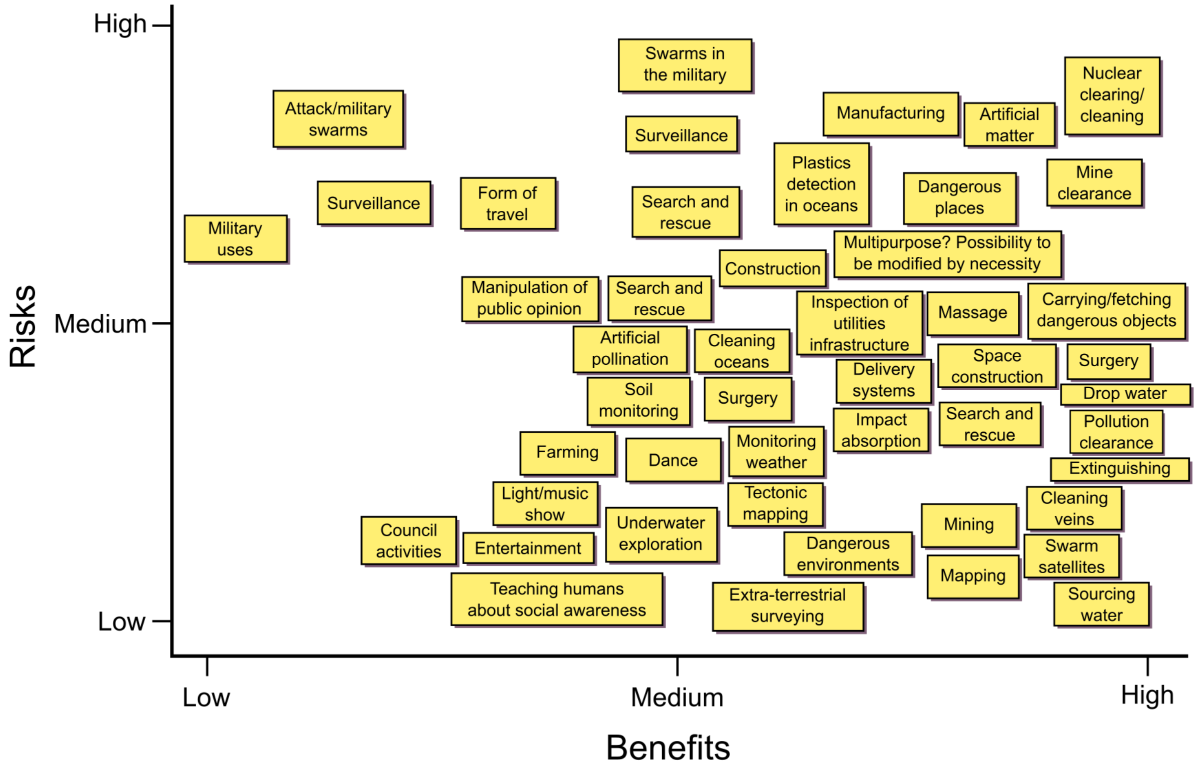


Figure 5.19: Benefits versus risks graph that participants of *Robot Swarms in our Cities* filled up with the possible uses of robot swarms that they came up with during the session.

tions were the same or similar to the ones proposed by the swarm robotics community [Brambilla et al., 2013; Şahin, 2005] (e.g. surveillance, search and rescue, exploration, mapping, dangerous environments, mine clearance), whereas others have barely been mentioned by the community (e.g. massage, form of travel). It is also worth noticing that most applications were placed in the medium-high benefits and low-medium risks, which offers a clear opportunity for development.

Answers to opportunities and concerns from participants were analysed and grouped into common themes. In terms of opportunities, I found the following common themes, with some examples of participants' comments per theme:

- **Improvement of quality of life:** drug delivery, replacing cells, companions.
- **Uses in dangerous/inaccessible environments:** cleaning oceans, risky construction locations, rescuing.
- **Adaptability:** replicable robots, gaps/breaks filling (e.g. utility pipes).
- **Removal of the human element:** labour-intensive activities such as farming or manufactures.
- **Efficiency:** mass virus inoculation, mass vaccination.

- **Enhanced entertainment:** immersive experiences, augmented reality.

With respect to participants' concerns, I found the following commonalities:

- **Misuse:** robot swarms hijacked, used as weapons, or for mass surveillance.
- **Sustainability:** robot swarms made of non-sustainable materials (e.g. non-degradable electronic components such as batteries), robot swarms becoming litter, harm to others (e.g. by being eaten by fish while cleaning plastic from oceans).
- **Removal of the human element:** robot swarms taking over jobs, people left behind.
- **Fear of the unknown:** hard to predict the future of robot swarms, and how they will be used.
- **Safety:** bias in data, robot swarms malfunctioning, or not enough robustness to failure.
- **Data security/privacy invasion:** exposure of personal data.
- **Ethics governance:** lack of regulation.
- **Lack of current applications:** robot swarms not fulfilling expectations, or becoming obsolete.

The final question had to do with requirements to make swarms beneficial in the future. I also grouped the common themes, that I present as follows with their corresponding explanation based on participants' comments:

- **Increase public confidence by showing benefits of robot swarms for society**: especially if uses of robot swarms are chosen by the communities, and they are not developed for private profit.
- **Increase public acceptance by educating society about technology:** explaining robot swarms is crucial to overcome public fears/concerns.
- **Transparency:** citizens should know what the robot swarm can/cannot do.
- **Safety:** extensive testing with thorough checks, and evidence of them.
- **Regulation:** to avoid misuse, or for risky developments.
- **Sustainable robot swarms:** research on sustainable materials (e.g. bio-degradable).

Among these themes, increasing public acceptance through education has historically been known as the *deficit model* of science communication, based on the assumption that society's lack of support for science and technology is directly related to their illiterateness on these topics [Miller, 1983]. However, recent research has found that this model does not hold true [Simis et al., 2016; Sturgis and Allum, 2004]. Instead, building trust and acceptance through community-based research in topics of interest to communities is an example of a much more effective approach.

### 5.3.3 Discussion

The use of a creative approach through the development of an innovative, educational escape room allowed participants to engage with the topic of swarm robotics in a unique way. This left participants more informed, allowing them to more effectively contribute to the conversation around how these technologies should be used in the future, as well as positively changing their perceptions towards swarm robotics. In particular, *Swarm Escape!* has been shown to increase players' knowledge of swarm robotics concepts, especially that there is communication involved among the robots. Furthermore, I have shown strong support for *Swarm Escape!* being the catalyst for unsure players to adopt a positive attitude towards working alongside robot swarms and their benefits for society. If researchers and developers of technology in general and robot swarms in particular are interested in user/society acceptance, artistic approaches such as immersive experiences seem to be a successful way to achieve this.

The group session was also shown to be a successful way to create a two-way impact through dialogue on both citizens and researchers, as the framework of mutual shaping aims to do [Šabanović, 2010]. Concretely, i) it prompted citizens to think of beneficial robot swarms applications, and ii) it provided swarm robotics researchers with insight about the main concerns about this technology from the public, as well as key actions to have beneficial robot swarms in the real world.

## 5.4 Concluding remarks

Both studies described in this chapter have been shown to have had a bidirectional impact: on participants (either potential users or the general public) and on swarm robotics researchers. After being informed about swarm robotics through a focus group in the former study, or through an escape room or a discussion session in the latter study, there was a common increase in acceptance of swarm robotics technology across participants. This was reflected in the question about the best number of robots to use in firefighting and rescuing in the study with fire brigades, and the question about how participants would feel if they were assisted by a robot swarm in the study with the general public. Indeed, the general public saw robot swarms for search and rescue as a good candidate for a future application. With respect to swarm robotics researchers, the use of an interactive public engagement approach in both studies widened the view on the possible applications of robot swarms. The general public advocated for tasks where humans might be at risk (e.g. nuclear cleaning, pollution clearance, search and rescue) or hard to do for humans (e.g. cleaning veins, impact absorption, tectonic mapping), as well as others not previously thought by the swarm robotics community (e.g. massage). In the case of fire brigades, their priority was information-gathering tasks (e.g. locating casualties, risk assessment, mapping). Therefore, it is important to identify the type of applications deemed as beneficial by users in particular and society in general for a successful deployment of robot swarms in the future.

With respect to concerns about swarm robotics, transparency and safety have been found in both studies to be crucial aspects for trust, also found in the robotics literature [Winfield and Jirotka, 2018]. The findings suggest that these aspects could be addressed by designing technology that is i) reliable through extensive testing, ii) environmentally sustainable with as less impact as possible, iii) applicable to the real world in a beneficial way, and iv) open through the inclusion of society in the research and development process. Indeed, firefighters expressed their desire to be included in the process from the very beginning.

This chapter has shown the power of a participatory design approach through mutual shaping to incorporate society's voices into the research and development process of technology, particularly into the field of swarm robotics. Swarm robotics technology is ready to move beyond laboratory conditions to the real world in a few years' time [Yang et al., 2018]. Hence, now is the perfect time for swarm robotics researchers to collaborate with society to define the role of robot swarms in our future cities–together. Indeed, research suggests that the act of engaging in a dialogue with communities about the science and technology that concerns them yields acceptance and trust by society, thus moving beyond the deficit model of science communication [Simis et al., 2016]. By embracing the culture of responsible research and innovation [Stilgoe et al., 2013], and ethical governance [Winfield and Jirotka, 2018]—through which ethical considerations found through dialogue with society are taken into account from the beginning of the research and development process—, swarm robotics researchers will be better able to realise the societal and economic benefits of this technology.

The following items should be considered for future work:

- Checking if findings from both studies still apply when engagement with society continues during the research and development process [Delmerico et al., 2019], especially after the inclusion of a more balanced representation of participants in terms of gender, background, race, etc.

- Advocating for the creation of the culture of ethical governance among institutions or companies doing research in swarm robotics (or developing robot swarms) to consider ethical aspects from the very beginning, as Winfield and Jirotka [2018] suggest, through an equal power relation between researchers and society, e.g. under the mutual shaping framework.

# Chapter 6

# Conclusion

This thesis merges developmental biology with morphogenetic engineering to develop the first demonstration of large swarms of real robots able to grow emergent, controllable and functional shapes in a completely self-organised fashion (without the need of a map of the target shape, coordinate system or preprogrammed seed robots). In the future, the algorithms presented in this thesis could be applied to solve problems in areas such as urban search and rescue (to explore a building on fire), utilities inspection (to detect cracks), space exploration (exploration of planets not reachable by humans) or nanomedicine (to fight cancer cells). For such application areas, the properties of emergence, adaptability, scalability and robustness of these algorithms might be highly beneficial. Furthermore, this thesis engaged society (i.e. potential users and the general public) with two studies using an interactive and participatory approach to create a bidirectional relationship with them, following the frameworks of mutual shaping and gamification. Under the umbrella of responsible research and innovation, these two studies are the first of their kind to understand attitudes, hopes, concerns and requirements from society with respect to the topic of swarm robotics in particular, giving concrete recommendations on how to advance in swarm robotics research to release its full, predicted potential in future real-world applications.

This chapter summarises the main contributions of this thesis, its limitations and future work on the topic of morphogenetic swarms and user/public engagement. In addition, a reflection on the potential impact on the environment, economy, society and culture is given.

## 6.1 Summary of contributions

The primary aim of this thesis was to show completely self-organised, bottom-up morphogenesis leading to emergent, adaptable, robust, controllable, and functional shapes in large robot swarms of real robots. The secondary aim was to design interactive engagement activities to create a bidirectional relationship between swarm robotics researchers and potential users and the general public. These aims led to the following contributions during this thesis:

### Morphogenesis in robot swarms

In chapter 3, the first demonstration of completely self-organised shape formation in large swarms of simple, real robots was given. The morphogenesis algorithm consisted of two main processes: patterning and migration. Inspiration came from the self-organised pattern formation nature of reaction-diffusion systems, as well as cell migration. Indeed, both processes have been proved to be crucial inductive and morphogenetic developmental mechanisms during the development of multi-cellular organisms. The patterning process was used to define regions across the swarm for growth to happen through the movement of robots at the edge of the swarm to such regions, hence extending them to create protrusions. By running patterning and migration alongside each other to have a large-scale, morphodynamic feedback loop, shapes were fully emergent, adaptable and robust. Twenty experiments with swarms of 300 Kilobots showed their ability to grow regular and organic morphologies starting from different initial configurations such as a circle or a rectangle, and their ability to self-repair when parts of their growing protrusions were removed, or the swarm was split in half. These results were published in the Science Robotics journal.

### Controllable morphogenesis

One of the drawbacks of the previous morphogenesis algorithm was the lack of controllability of shapes. Chapter 4 combined patterning, migration and local gradients to develop a more controllable morphogenesis algorithm, hence being a step closer to the aims of the field of morphogenetic engineering. Inspiration came from one of the hypotheses given by Green and Sharpe [2015], which states that reaction-diffusion and positional information could indeed be working together during the development of biological organisms. For this extension, patterning and migration were decoupled from each other to improve controllability. Patterning was used to establish initial regions across the swarm for growth to happen. As opposed to the previous morphogenesis algorithm, robots maintained such regions through signals that created a gradient in each of them. Controllability of the morphogenesis process was then achieved by having different morphogenesis variables that influenced each gradient, hence the regions of growth. Simulation experiments showed the wider morphospace created with this approach, as compared with the morphogenesis algorithm using reaction-diffusion and migration alone. Furthermore, over 2000 simulations were run to show the scalability properties of swarms using this approach. Two extra simulations showed the ability of swarms to get around a simple, static obstacle. Five experiments with swarms of 300 Kilobots also showed the emergence of shapes and their robustness to perturbations when protrusions were cut off. These results were published in the IEEE Robotics and Automation Letters journal, and presented at IEEE IROS 2019.

### Functional morphogenesis

As shown at the end of chapter 4, the controllable morphogenesis algorithm was extended for functionality, inspired by the complex behaviours seen in *Physarum polycephalum*. For example, this unicellular organism modifies its shape to forage by branching out protoplasmic tubes. For this extension, when the robots in the swarm found objects of potential interest while exploring the environment through controllable morphogenesis, they sent a hop-count-based signal across the swarm. All robots then worked out whether they were in any of the shortest paths between such objects of potential interest. When they were, they lighted up their LEDs as a signal for users (i.e. swarm-guided navigation [Brambilla et al., 2013]). Experiments in simulation showed that swarms managed to explore the environment through morphogenesis and the lighting up of the shortest path between two objects of potential interest.

### Engagement with users

In chapter 5, a study with fire brigades—potential users of the morphogenetic swarm robotics technology developed in this thesis—was presented following the principles of mutual shaping through participatory engagement. Indeed, this study contributed to filling the gap in the literature with respect to the lack of user studies in swarm robotics, especially regarding user needs, attitudes and concerns. This study was crucial, as swarm robotics has been categorised as an underpinning technology with a potential impact on all application areas of robotics, and it has been predicted to have considerable advances and impact during this decade.

The study with fire brigades shed light on their needs, attitudes and concerns about robot swarms. Fire brigades were generally positive about the use of robot swarms to assist firefighters in fire and rescue missions. They expressed their desire to have robot swarms performing information-gathering tasks (e.g. mapping, locating casualties, creating communication links, etc.) in a semi-autonomous way, as opposed to swarms carrying out all the decision-making process. In addition, this study identified fire brigades' requirements for them to trust robot swarms, with these being related to increased transparency, accountability, reliability, safety, and ease of operation. Indeed, trust is essential for the successful implementation of swarm robotics systems in real-world applications, hence unleashing their economic and societal benefits [Winfield and Jirotka, 2018]. Finally, mutual shaping was shown to be a suitable framework to establish a bidirectional engagement between researchers and users from the beginning of the research and development process—as expressed by fire brigades—, hence benefiting both sides. These results were published in the Special Issue on Designing Self-Organization in the Physical Realm of Frontiers in Robotics and AI - Multi Robot Systems journal.

### Engagement with the general public

An educational, portable escape room named *Swarm Escape!* was developed in chapter 5 as a way to engage the general public in the topic of swarm robotics. Apart from communicating the research done in this thesis, it is important to educate society about future uses of swarm robotics, as fire brigades expressed in the previous study. The escape room was built around a fictional scenario, with players becoming swarm robotics researchers to help emergency services mitigate a pollution crisis, as inspired by the potential search and rescue application of the morphogenesis algorithms developed in this thesis. Puzzles featured videos and robots used in the morphogenesis part of this thesis. The escape room was shown to increase players' knowledge of swarm robotics concepts, and to make a positive impact on unsure players.

In addition, chapter 5 detailed how the use of mutual shaping to structure a group discussion session named *Robot Swarms in our Cities* also impacted both the general public and swarm robotics researchers. In particular, it prompted citizens to think of beneficial robot swarms applications. Examples of these were search and rescue, pollution cleaning, surgery, construction, tectonic mapping, or swarm satellites. This session also provided swarm robotics researchers with insight about the main concerns about this technology from the public, as well as key actions to deploy beneficial robot swarms in the real world. Concerns were related to misuse, sustainability, removal of the human element, fear of the unknown, safety, privacy, ethics and lack of applications. Participants suggested increasing transparency, safety, regulation, sustainability, visibility and understanding as ways towards successfully deployment of robot swarms in society.

## 6.2 Limitations and future work

This thesis successfully achieved its main goals of developing completely self-organised, controllable and functional morphogenesis algorithms for large swarms of real robots, and engaging in a two-way dialogue with potential users of this technology and the general public. However, there are certain limitations and scope for improvement, which are summarised below:

- Instabilities in the reaction-diffusion system implemented in chapter 3 were one of the the main drivers to seek controllability in chapter 4. However, their role in morphogenesis should be studied better, as they could be a great ally to flexibility to the functional morphogenesis algorithms. If instabilities could be triggered when required, robot swarms could use them to overcome situations when they may have become stuck.

- Maintaining connectivity within the swarm is crucial for applications such as search and rescue. Indeed, the functional morphogenesis algorithm would not be able to find the shortest path between objects of potential interest if there was a permanent loss of connectivity in the swarm. In the algorithms developed in this thesis, shape growth was achieved through robots migrating from one area to another area in the swarm by following the edge of the swarm. However, there is no mechanism in place to prevent the swarm from splitting into smaller swarms. Indeed, if experiments were run for a longer time, small, disconnected islands of robots may appear. As a consequence, it is important to improve the morphogenesis algorithms to stop this situation from happening.

- Although the controllable morphogenesis algorithm presented in chapter 4 has been shown to increase the type of shapes that swarms can create by means of using different combinations of morphogenesis variables, there were other regions in the morphospace that swarms could not apparently reach. This is still a burden that might limit the controllability of the approach if compared with state-of-the-art algorithms that control the swarm shapes by means of providing a map of the shape to create. Therefore, it is important to expand the range of controllable shapes without losing self-organisation. For that, a full characterisation on the limits of controllability using morphogenesis variables and local gradients would be highly beneficial. In addition, participants from fire brigades expressed their preference for semi-autonomous control over swarms. Therefore, further research on how to control the type of morphologies that swarms create without explicitly giving the swarm the shape would contribute towards achieving such semi-autonomy.

- The morphogenesis algorithms developed in this thesis have only been tested in laboratory conditions. In the case of the functional morphogenesis algorithm, time only allowed testing it in simulation. If morphogenetic robot swarms are to be deployed for future real-world applications such as search and rescue, it is important to design scalable swarm robotics hardware that can be used for such real-world applications. Then, the morphogenesis algorithms proposed in this thesis can be tested and improved to tailor them to the needs of end users. For instance, exploration through morphogenesis should be fast, as fire brigades tend to have little time to gather information before action, as they expressed.

- With respect to the engagement of society (users, stakeholders and the general public), this thesis engaged as a one-off experience, i.e. participants were only recruited once for the focus groups, escape room or group discussion session. This was particularly useful to identify the next steps for swarm robotics to become a successful reality in future applications by means of having a dialogue with society through participative and interactive approaches. However, if societal acceptance and trust in swarm robotics are sought, this can only be done through the inclusion of society in the complete research and development process, from the beginning. This is actually a desire from the fire brigades which participated in the study, as detailed in chapter 5. Furthermore, a higher number of participants with a wider range of gender, background, race, etc. should be engaged to have a better generalisation of results. To unleash the economic and societal benefits predicted for swarm robotics, it is crucial to engage society (e.g. communities who will be affected by the technology, potential users, etc.) from the very beginning, in a continuous, equitable and bidirectional manner. This is a key part of the culture of ethical governance and responsible research and innovation that research institutions should embrace.

## 6.3 Environmental, economic, societal and cultural future impact

The advances made in this thesis bring swarm robotics a step closer to becoming a real solution for several applications in the future. Therefore, it is important to take a step back, look at the bigger picture, and reflect on the possible consequences for the environment, the economy, society and culture.

### Environment

By definition, swarm robotics implies a large number of robots. In this sense, the concept of scalability plays a huge role in the success of swarm robotics approaches—if the algorithms are not scalable, they may not be suitable for a swarm robotics approach. However, as the number of robots scales up, so it does their environmental impact due to their electronic components, for example. This was indeed stressed out by the general public in the *Robot Swarms in our Cities* session. Solutions could be designing swarm robotics platforms powered through renewable energy, and/or biodegradable robot swarms that do not cause any harm to the environment [Rossiter et al., 2016].

### Economy and society

As described in chapter 5, potential users and the general public are positive about using swarm robotics for future applications. Indeed, there were many uses suggested by society with foreseeable benefits, hence positively affecting the economy. This means there is scope for businesses to invest in some of those applications, because they show a preliminary level of acceptance by society. However, it is important to continually check whether that acceptance still applies, or even better, to include society in the research and development process to guarantee successful development of swarm robotics technology—as this thesis argues. To increase acceptance and trust in swarm robotics technology, transparency and safety have to be at the core of it, as deemed by society. As also demanded by society, this could be achieved through standards and regulation.

### Culture

Robotics and artificial intelligence are driving the $4^{th}$ industrial revolution [Skilton and Hovsepian, 2017]. As swarm robotics has been predicted to have a huge impact on all application areas of robotics [Yang et al., 2018], it is also part of this technological revolution. The difference with the previous revolutions is perhaps time. This $4^{th}$ industrial revolution is developing faster than its predecessors as a result of the improvement in computing power and data. This rhythm of development might cause a big impact in our culture by society having to quickly adapt to new advances to avoid becoming obsolete. Or worse: society could start to distrust such advances simply because technology developers are too focused on development *per se* rather than creating meaningful technology hand in hand with society. By practising *slow science* [Stengers, 2018], we, society, can have time to digest and drive the technological progress that we want–together. Just because we can develop the technology, it does not mean it is ethical and beneficial for us. Reflectivity is as important (or even more) as progress. We, researchers, have the power to reclaim its important role. We can set an example.

# Appendix: Battery and noise improvements on the Kilobot firmware

## Modified Kilobot firmware 2.0 to filter out noise

During experiments in this thesis, I realised that some Kilobots went to *SLEEPING* or *IDLE* mode when running. I believe this happens due to i) background infrared noise that sometimes interferes with the Kilobots, and/or ii) if the message from a Kilobot gets corrupted. When this happens, the robots might interpret the signal/message as one coming from the overhead controller commanding them to go to *SLEEPING* or *IDLE* state.

I have modified Firmware 2.0[1] to filter out this noise by adding a counter for the number of consecutive times that the robot receives a message with type ≥ 128 (user messages must have a type < 128) in the *RUNNING* state. When it reaches a certain threshold (10 consecutive messages), it then carries out the command. After this modification to the firmware, the Kilobots take a bit more time to process the signal from the controller when running (between 0.5 and 2 seconds delay), but none of them go the *SLEEPING* or *IDLE* while running.

The modified library can be found here[2]. Concretely, a new constant and variable named *THRESHOLD_N_COMMANDS* and *n_commands_received*, respectively, have been added to *kilolib.c*, and the *process_message* function has been properly adapted. The modified code is the following:

```
// Parameter for IR filter
#define THRESHOLD_N_COMMANDS 10

// Number of consecutive commands received from the programmer (or noise)
uint8_t n_commands_received;

static inline void process_message() {
    AddressPointer_t reset = (AddressPointer_t)0x0000, bootload =
        (AddressPointer_t)0x7000;
    calibmsg_t *calibmsg = (calibmsg_t*)&rx_msg.data;
    if (rx_msg.type < BOOT) {
```

[1] https://github.com/acornejo/kilolib

[2] https://github.com/Danixk/kilolib

```
        kilo_message_rx(&rx_msg, &rx_dist);
        n_commands_received = 0;
        return;
    }

    // Doesn't respond to other signals apart from neighbors while running if a few
        received
    if(kilo_state == RUNNING && n_commands_received < THRESHOLD_N_COMMANDS){
        n_commands_received++;
        return;
    }

    // In RUNNING state, it has to receive at least THREHOLD_N_COMMANDS continuous
        commands to react
    else if( (kilo_state == RUNNING && n_commands_received == THRESHOLD_N_COMMANDS)
        || (kilo_state != RUNNING)){

        n_commands_received = 0;

        if (rx_msg.type != READUID && rx_msg.type != RUN && rx_msg.type != CALIB)
            motors_off();
        switch (rx_msg.type) {
            case BOOT:
                tx_timer_off();
                bootload();
                break;
            case RESET:
                reset();
                break;
            case SLEEP:
                kilo_state = SLEEPING;
                break;
            case WAKEUP:
                kilo_state = IDLE;
                break;
            case CHARGE:
                kilo_state = CHARGING;
                break;
            case VOLTAGE:
                kilo_state = BATTERY;
                break;
            case RUN:
                if (kilo_state != SETUP && kilo_state != RUNNING) {
```

```
                motors_on();
                kilo_state = SETUP;
            }
            break;
        case CALIB:
            switch(calibmsg->mode) {
                case CALIB_SAVE:
                    if (kilo_state == MOVING) {
                        eeprom_write_byte(EEPROM_UID, kilo_uid&0xFF);
                        eeprom_write_byte(EEPROM_UID+1, (kilo_uid>>8)&0xFF);
                        eeprom_write_byte(EEPROM_LEFT_ROTATE, kilo_turn_left);
                        eeprom_write_byte(EEPROM_RIGHT_ROTATE, kilo_turn_right);
                        eeprom_write_byte(EEPROM_LEFT_STRAIGHT, kilo_straight_left);
                        eeprom_write_byte(EEPROM_RIGHT_STRAIGHT,
                            kilo_straight_right);
                        motors_off();
                        kilo_state = IDLE;
                    }
                    break;
                case CALIB_UID:
                    kilo_uid = calibmsg->uid;
                    cur_motion = MOVE_STOP;
                    break;
                case CALIB_TURN_LEFT:
                    if (cur_motion != MOVE_LEFT || kilo_turn_left !=
                        calibmsg->turn_left) {
                        prev_motion = MOVE_STOP;
                        cur_motion = MOVE_LEFT;
                        kilo_turn_left = calibmsg->turn_left;
                    }
                    break;
                case CALIB_TURN_RIGHT:
                    if (cur_motion != MOVE_RIGHT || kilo_turn_right !=
                        calibmsg->turn_right) {
                        prev_motion = MOVE_STOP;
                        cur_motion = MOVE_RIGHT;
                        kilo_turn_right = calibmsg->turn_right;
                    }
                    break;
                case CALIB_STRAIGHT:
                    if (cur_motion != MOVE_STRAIGHT || kilo_straight_right !=
                        calibmsg->straight_right || kilo_straight_left !=
                        calibmsg->straight_left) {
```

```
                    prev_motion = MOVE_STOP;
                    cur_motion = MOVE_STRAIGHT;
                    kilo_straight_left = calibmsg->straight_left;
                    kilo_straight_right = calibmsg->straight_right;
                }
                break;
            }
            if (calibmsg->mode != CALIB_SAVE && kilo_state != MOVING) {
                motors_on();
                kilo_state = MOVING;
            }
            break;
        case READUID:
            if (kilo_state != MOVING) {
                motors_on();
                set_color(RGB(0,0,0));
                prev_motion = cur_motion = MOVE_STOP;
                kilo_state = MOVING;
            }

            if (kilo_uid&(1<<rx_msg.data[0]))
                cur_motion = MOVE_LEFT;
            else
                cur_motion = MOVE_STOP;
            break;
        default:
            break;
    }
  }
}
```

## Modified Kilobot firmware 2.0 to protect the battery

I also realised that Firmware 2.0 does not have any battery protection. This means that if Kilobots are left switched on indefinitely, their batteries will drain until the point when they cannot be recharged anymore, and a new battery is needed. To solve this issue, a hardware solution would be ideal. However, this would require remanufacturing the robots, which is time consuming. Therefore, I have modified Firmware 2.0 to add a software solution to this. The modified library can be found here too[3].

[3] https://github.com/Danixk/kilolib

The solution I have implemented consists of the robots estimating the voltage frequently during the *RUNNING*, *IDLE* and *SLEEPING* state. If they measure a low voltage value (3.2V) for a certain number of consecutive times, they will switch to the *SLEEPING* state, i.e. to low-power mode to stop running and save battery. Below is a detailed explanation of all the changes.

The following constants and variables have been added to *kilolib.c*:

```
// Parameters for battery protection
#define N_READINGS_VOLTAGE 10
#define N_REPETITIONS_LOW_VOLTAGE 5
#define LOW_VOLTAGE_THRESHOLD 547   // Equivalent to 3.2V
#define HIGH_VOLTAGE_THRESHOLD 716  // Equivalent to 4.2V

int16_t voltage = -1;               // Voltage estimation
uint8_t is_in_low_voltage = 0;      // Number of consecutive times it reads a
    voltage below the low-voltage threshold
uint8_t counter_ticks_for_voltage = 0; // To estimate voltage every 255 ticks (~8
    seconds)
```

The Kilobots estimate the voltage approximately every 8 seconds in the *RUNINNG* state (less time might interfere with distance estimation) using a 1-byte counter that is increased every *kilo_tick* (255 *kilo_ticks* ≘ 8 seconds), constantly in the *IDLE* state, and every time they wake up in the *SLEEPING* state. The voltage is not estimated in the *BATTERY* state or the rest (*SETUP*, *CHARGING*, *MOVING*) because the user is supposed to be supervising or interacting with the robots. To increase *counter_ticks_for_voltage* every *kilo_ticks*, the *Timer0* interrupt function has been modified as follows:

```
/**
 * Timer0 interrupt.
 * Used to send messages every kilo_tx_period ticks.
 */
ISR(TIMER0_COMPA_vect) {
    tx_clock += tx_increment;
    tx_increment = 0xFF;
    OCR0A = tx_increment;
    kilo_ticks++;

    // Increments counter_ticks_for_voltage by 1 tick until 255
    if(kilo_state == RUNNING && counter_ticks_for_voltage < 255)
        counter_ticks_for_voltage++;

    if(!rx_busy && tx_clock>kilo_tx_period && kilo_state == RUNNING) {
        message_t *msg = kilo_message_tx();
        if (msg) {
```

```
            if (message_send(msg)) {
                kilo_message_tx_success();
                tx_clock = 0;
            } else {
                tx_increment = rand()&0xFF;
                OCR0A = tx_increment;
            }
        }
    }
}
```

To estimate the voltage, a new function *estimate_voltage* has been added, accepting the binary parameter *trigger_high_gain* as input. This function takes *N_READINGS_VOLTAGE* consecutive measurements of the voltage, and returns the maximum of them. This acts as a filter for measurements of -1 from *get_voltage()* but also as a filter for the first voltage reading in case the last time this function was executed with *trigger_high_gain* equal to 1 (*true*). The reason is that most of the functions using the analog-to-digital converter (*get_ambientlight*, *get_temperature*, *rand_hard* and *estimate_distance*) need to measure high gain to give a correct value (that is why they execute *adc_trigger_high_gain()* at the end). However, *get_voltage()* gives a wrong value of the voltage if measuring high gain (that is why *adc_trigger_high_gain()* is commented out at the end). In the case of the *RUNNING* state, the user might call any of the functions using the analog-to-digital converter that need to measure high gain, so those functions will leave high gain triggered. If voltage is then read, the value will be wrong. However, next time the voltage is read, the value will be correct as long as none of the other functions is executed in between. To then allow the other functions to give correct values, high gain must be triggered after the last reading of the voltage in the new *estimate_voltage* function. However, this is only applicable to the *RUNNING* state, because in the other states where the voltage is estimated (*SLEEPING* and *IDLE*), the functions using the analog-to-digital converter are not used. Therefore, *estimate_voltage(1)* is executed when estimating the voltage during the *RUNNING* state, and *estimate_voltage(0)* in *SLEEPING* and *IDLE*. The code for the *estimate_voltage* function is the following:

```
int16_t estimate_voltage(uint8_t trigger_high_gain){

    uint8_t i;
    int16_t voltage;
    int16_t estimated_voltage = -1;

    for(i = 0; i < N_READINGS_VOLTAGE; i++){
        voltage = get_voltage();
        if(voltage <= HIGH_VOLTAGE_THRESHOLD && voltage > estimated_voltage){
            estimated_voltage = voltage;
```

```
        }
    }

    if(trigger_high_gain){
        cli();
        adc_trigger_high_gain();
        sei();
    }

    return estimated_voltage;
}
```

Finally, the *kilo_start* function had to be modified to include voltage estimation. Furthermore, the LED colour flashed in the *SLEEPING* and *IDLE* states has been slightly modified to show the corresponding colour of the battery level (same scale as in the *BATTERY* state). This allows easy identification of the level of charge of the battery without sending them to the *BATTERY* state (particularly useful when sleeping, because they do not have to be waken up to know how much battery they have got left). In case they switch to *SLEEPING* state due to low battery (or because they detect low battery while sleeping), they will flash red for 1 second to signal their battery is in danger. The modified *kilo_start* function is the following:

```
void kilo_start(void (*setup)(void), void (*loop)(void)) {
    uint8_t has_setup = 0;
    while (1) {
        switch(kilo_state) {
            case SLEEPING:
                cli();
                acomp_off();
                adc_off();
                ports_off();
                wdt_enable(WDTO_8S);
                WDTCSR |= (1<<WDIE);
                set_sleep_mode(SLEEP_MODE_PWR_DOWN);
                cli();
                sleep_enable();
                sei();
                sleep_cpu();
                sleep_disable();
                sei();
                rx_busy = 0;
                ports_on();
                adc_on();
                _delay_us(300);
```

```
        acomp_on();

        if(is_in_low_voltage == N_REPETITIONS_LOW_VOLTAGE){ // Low battery
            set_color(RGB(3,0,0));
            _delay_ms(1000);
            set_color(RGB(0,0,0));
        }
        else{
            voltage = estimate_voltage(0);
            if(voltage != -1 && voltage < LOW_VOLTAGE_THRESHOLD){
                is_in_low_voltage++;
            }
            else if(voltage != -1){
                is_in_low_voltage = 0;
            }
            if(voltage > 682)
                set_color(RGB(0,3,0));
            else if(voltage > 648)
                set_color(RGB(0,0,3));
            else if(voltage > 614)
                set_color(RGB(3,3,0));
            else if(voltage != -1)
                set_color(RGB(3,0,0));
            else
                set_color(RGB(3,3,3));

            _delay_ms(10);
            if (rx_busy) {
                set_color(RGB(0,3,0));
                _delay_ms(100);
            }
            set_color(RGB(0,0,0));
        }
        break;
    case IDLE:
        if(rx_busy){
            set_color(RGB(0,3,0));
            _delay_ms(1);
            set_color(RGB(0,0,0));
            _delay_ms(200);
        }
        else if(is_in_low_voltage < N_REPETITIONS_LOW_VOLTAGE){
            voltage = estimate_voltage(0);
```

```
            if(voltage != -1 && voltage < LOW_VOLTAGE_THRESHOLD){
                is_in_low_voltage++;
            }
            else if(voltage != -1){
                is_in_low_voltage = 0;
            }

            if(voltage > 682)
                set_color(RGB(0,3,0));
            else if(voltage > 648)
                set_color(RGB(0,0,3));
            else if(voltage > 614)
                set_color(RGB(3,3,0));
            else if(voltage != -1)
                set_color(RGB(3,0,0));
            else
                set_color(RGB(3,3,3));

            _delay_ms(1);
            set_color(RGB(0,0,0));
            _delay_ms(200);
        }
        else{
            kilo_state = SLEEPING;
        }
        break;
    case BATTERY:
        voltage = get_voltage();
        is_in_low_voltage = 0;
        if(voltage > 682)
            set_color(RGB(0,3,0));
        else if(voltage > 648)
            set_color(RGB(0,0,3));
        else if(voltage > 614)
            set_color(RGB(3,3,0));
        else
            set_color(RGB(3,0,0));
        break;
    case CHARGING:
        is_in_low_voltage = 0;
        if (is_charging()) {
            set_color(RGB(1,0,0));
            _delay_ms(1);
```

```
            set_color(RGB(0,0,0));
            _delay_ms(200);
        } else
            set_color(RGB(0,0,0));
        break;
    case SETUP:
        if (!has_setup) {
            setup();
            has_setup = 1;
        }
        is_in_low_voltage = 0;
        kilo_state = RUNNING;
    case RUNNING:
        if(rx_busy){
            loop();
        }
        else if(is_in_low_voltage < N_REPETITIONS_LOW_VOLTAGE){
            // Estimate voltage every 8 seconds, approximately
            cli(); // Disable interrupts to read or write variable
                counter_ticks_for_voltage
            if(counter_ticks_for_voltage == 255){
                counter_ticks_for_voltage = 0;
                sei(); // Enable interrupts
                voltage = estimate_voltage(1);
                if(voltage != -1 && voltage < LOW_VOLTAGE_THRESHOLD){
                    is_in_low_voltage++;
                }
                else if(voltage != -1){
                    is_in_low_voltage = 0;
                }
            }
            else
                sei(); // Enable interrupts
            loop();
        }
        else{
            kilo_state = SLEEPING;
        }
        break;
    case MOVING:
        is_in_low_voltage = 0;
        if (cur_motion == MOVE_STOP) {
            set_motors(0,0);
```

```
                prev_motion = MOVE_STOP;
            } else {
                if (cur_motion != prev_motion) {
                    prev_motion = cur_motion;
                    if (cur_motion == MOVE_LEFT) {
                        set_motors(0xFF, 0);
                        _delay_ms(15);
                        set_motors(kilo_turn_left, 0);
                    } else if (cur_motion == MOVE_RIGHT) {
                        set_motors(0, 0xFF);
                        _delay_ms(15);
                        set_motors(0, kilo_turn_right);
                    } else {
                        set_motors(0, 0xFF);
                        set_motors(0xFF, 0xFF);
                        _delay_ms(15);
                        set_motors(kilo_straight_left, kilo_straight_right);
                    }
                }
            }
            break;
        }
    }
}
```

So far, all tests at the Bristol Robotics Laboratory have been successful in terms of protecting the Kilobots by sending them to sleep when the battery is low. In case of faulty/bad batteries, robots might go to sleep well before the voltage has dropped. However, this tells the battery is indeed faulty. I believe this protection mechanism is worth having even if some robots are sent to sleep prematurely, as it avoids the loss of batteries. More tests from other teams would be very valuable.

# Bibliography


Abelson, H., Allen, D., Coore, D., Hanson, C., Homsy, G., Knight Jr, T. F., Nagpal, R., Rauch, E., Sussman, G. J., and Weiss, R. (2000). Amorphous computing. *Communications of the ACM*, 43(5):74–82.

Adamatzky, A. (2016). *Advances in Physarum machines: Sensing and computing with slime mould*, volume 21. Springer.

Agresti, A. (2003). *Categorical data analysis*, volume 482. John Wiley & Sons.

Akam, M. (1989). Hox and hom: Homologous gene clusters in insects and vertebrates. *Cell*, 57(3):347–349.

Alcantara, F. and Monk, M. (1974). Signal propagation during aggregation in the slime mould dictyostelium discoideum. *Microbiology*, 85(2):321–334.

Alvarado, C. R. and Stephenson, S. L. (2017). *Myxomycetes: Biology, Systematics, Biogeography and Ecology*. Academic Press.

Amack, J. D. and Manning, M. L. (2012). Knowing the boundaries: Extending the differential adhesion hypothesis in embryonic cell sorting. *Science*, 338(6104):212–215.

Anderson, C., Theraulaz, G., and Deneubourg, J.-L. (2002). Self-assemblages in insect societies. *Insectes Sociaux*, 49(2):99–110.

Angermueller, C., Pärnamaa, T., Parts, L., and Stegle, O. (2016). Deep learning for computational biology. *Molecular Systems Biology*, 12(7).

Annadhason, A. (2012). Methods of fractal dimension computation. *IRACST—International Journal of Computer Science and Information Technology & Security (IJCSITS) vol*, 2.

Arnal, M., Macías García, J., and Duarte Tosso, I. (2019). Escape rooms as a way to teach magnitudes and measure in degrees in education. In *Proceedings of New Perspectives in Science Education*, pages 1–4.

Attygalle, A. B. and Morgan, E. D. (1985). Ant trail pheromones. In *Advances in Insect Physiology*, volume 18, pages 1–30. Elsevier.

Bai, L., Breen, D. E., and Eyiyurekli, M. (2008). An emergent system for self-aligning and self-organizing shape primitives. In *2008 Second IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO)*, pages 445–454.

Ballaro, B., Reas, P., and Tegolo, D. (2002). Elliptical fourier descriptors for shape retrieval in biological images. In *International Conference on Electronics, Control & Signal Processing. SG*.

Baroux, C. and Grossniklaus, U. (2015). The maternal-to-zygotic transition in flowering plants: Evidence, mechanisms, and plasticity. In *Current Topics in Developmental Biology*, volume 113, pages 351–371. Elsevier.

Becker, A., Ertel, C., and McLurkin, J. (2014). Crowdsourcing swarm manipulation experiments: A massive online user study with large swarms of simple robots. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2825–2830. IEEE.

Beekman, M. and Latty, T. (2015). Brainless but multi-headed: Decision making by the acellular slime mould physarum polycephalum. *Journal of Molecular Biology*, 427(23):3734 – 3743. Cooperative Behaviour in Microbial Communities.

Beltrame, G., Merlo, E., Panerati, J., and Pinciroli, C. (2018). Engineering safety in swarm robotics. In *Proceedings of the 1st International Workshop on Robotics Software Engineering*, RoSE ’18, page 36–39, New York, NY, USA. Association for Computing Machinery.

Beni, G. (2005). From swarm intelligence to swarm robotics. In Şahin, E. and Spears, W. M., editors, *Swarm Robotics. Lecture Notes in Computer Science*, volume 3342, pages 1–9, Berlin, Heidelberg. Springer Berlin Heidelberg.

Berg, J., Lottermoser, A., Richter, C., and Reinhart, G. (2019). Human-robot-interaction for mobile industrial robot teams. *Procedia CIRP*, 79:614 – 619. 12th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 18-20 July 2018, Gulf of Naples, Italy.

Bhapkar, V. P. (1966). A note on the equivalence of two test criteria for hypotheses in categorical data. *Journal of the American Statistical Association*, 61(313):228–235.

Bissonette, J. A. (2012). *Wildlife and landscape ecology: Effects of pattern and scale*. Springer Science & Business Media.

Bjerknes, J. D. and Winfield, A. F. T. (2013). On fault tolerance and scalability of swarm robotic systems. In *Distributed Autonomous Robotic Systems*, volume 83, pages 431–444. Springer.

Boczkowski, P. J. (1999). Mutual shaping of users and technologies in a national virtual community. *Journal of Communication*, 49(2):86–108.

Boisseau, R. P., Vogel, D., and Dussutour, A. (2016). Habituation in non-neural organisms: Evidence from slime moulds. *Proceedings of the Royal Society of London B: Biological Sciences*, 283(1829).

Bonabeau, E., Dorigo, M., and Theraulaz, G. (1999). *Swarm intelligence: From natural to artificial systems*. Number 1. Oxford university press.

Bonani, M., Longchamp, V., Magnenat, S., Rétornaz, P., Burnier, D., Roulet, G., Vaussard, F., Bleuler, H., and Mondada, F. (2010). The marxbot, a miniature mobile robot opening new perspectives for the collective-robotic research. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4187–4193. IEEE.

Bonner, J. T. and Savage, L. (1947). Evidence for the formation of cell aggregates by chemotaxis in the development of the slime mold dictyostelium discoideum. *Journal of Experimental Zoology*, 106(1):1–26.

Bowerman, B. and Shelton, C. A. (1999). Cell polarity in the early caenorhabditis elegans embryo. *Current Opinion in Genetics & Development*, 9(4):390–395.

Brambilla, M., Ferrante, E., Birattari, M., and Dorigo, M. (2013). Swarm robotics: A review from the swarm engineering perspective. *Swarm Intelligence*, 7(1):1–41.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

Briscoe, J. (2019). Understanding pattern formation in embryos: Experiment, theory, and simulation. *Journal of Computational Biology*, 26(7):696–702.

Briscoe, J. and Small, S. (2015). Morphogen rules: Design principles of gradient-mediated embryo patterning. *Development*, 142(23):3996–4009.

Bucchi, M. and Trench, B. (2014). Publics and their participation in science and technology: Changing roles, blurring boundaries. In *Routledge Handbook of Public Communication of Science and Technology*, pages 141–155. Routledge.

Bullock, S., Ladley, D., and Kerby, M. (2012). Wasps, termites, and waspmites: Distinguishing competence from performance in collective construction. *Artificial Life*, 18(3):267–290.

Camazine, S., Deneubourg, J.-L., Franks, N. R., Sneyd, J., Bonabeau, E., and Theraulaz, G. (2003). *Self-organization in biological systems*, volume 7. Princeton university press.

Carrillo-Zapata, D., Lee, C., Digumarti, K. M., Hauert, S., and Boushel, C. (2020). Lessons from delivering a stem workshop using educational robots given language limitations. In Merdan, M., Lepuschitz, W., Koppensteiner, G., Balogh, R., and Obdržálek, D., editors, *Robotics in Education*, pages 284–295, Cham. Springer International Publishing.

Chan, S.-K. and Struhl, G. (1997). Sequence-specific RNA binding by bicoid. *Nature*, 388(6643):634–634.

Chen, Y. and Sundaram, H. (2005). Estimating complexity of 2D shapes. In *2005 IEEE 7th Workshop on Multimedia Signal Processing*, pages 1–4. IEEE.

Chen, Y. and Zhao, X. (1998). Shaping limbs by apoptosis. *Journal of Experimental Zoology*, 282(6):691–702.

Cheng, J., Cheng, W., and Nagpal, R. (2005). Robust and self-repairing formation control for swarms of mobile agents. In *National Conference on Artificial Intelligence (AAAI)*, volume 5.

Chiang, C., Litingtung, Y., Lee, E., Young, K. E., Corden, J. L., Westphal, H., and Beachy, P. A. (1996). Cyclopia and defective axial patterning in mice lacking sonic hedgehog gene function. *Nature*, 383(6599):407–413.

Clarke, S., Peel, D. J., Arnab, S., Morini, L., Keegan, H., and Wood, O. (2017). Escaped: A framework for creating educational escape rooms and interactive games for higher/further education. *International Journal of Serious Games*, 4(3):73–86.

Couceiro, M. S., Figueiredo, C. M., Luz, J. M. A., Ferreira, N. M., and Rocha, R. P. (2012). A low-cost educational platform for swarm robotics. *International Journal of Robots, Education & Art*, 2(1).

Couture-Beil, A., Vaughan, R. T., and Mori, G. (2010). Selecting and commanding individual robots in a vision-based multi-robot system. In *2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 355–356. IEEE.

Davidson, E. H. (2010). *The regulatory genome: Gene regulatory networks in development and evolution*. Elsevier.

Davies, J. (2013). *Mechanisms of morphogenesis*. Academic Press.

Delmerico, J., Mintchev, S., Giusti, A., Gromov, B., Melo, K., Horvat, T., Cadena, C., Hutter, M., Ijspeert, A., Floreano, D., Gambardella, L. M., Siegwart, R., and Scaramuzza, D. (2019). The current state and future outlook of rescue robotics. *Journal of Field Robotics*, 36(7):1171–1191.

Dicheva, D., Dichev, C., Agre, G., and Angelova, G. (2015). Gamification in education: A systematic mapping study. *Journal of Educational Technology & Society*, 18(3).

Digumarti, K. M., Trimmer, B., Conn, A. T., and Rossiter, J. (2019). Quantifying dynamic shapes in soft morphologies. *Soft Robotics*, 6(6):733–744. PMID: 31314665.

Dixon, C., Winfield, A. F. T., Fisher, M., and Zeng, C. (2012). Towards temporal verification of swarm robotic systems. *Robotics and Autonomous Systems*, 60(11):1429–1441.

Dorigo, M., Floreano, D., Gambardella, L. M., Mondada, F., Nolfi, S., Baaboura, T., Birattari, M., Bonani, M., Brambilla, M., Brutschy, A., Burnier, D., Campo, A., Christensen, A. L., Decugniere, A., Di Caro, G., Ducatelle, F., Ferrante, E., Forster, A., Gonzales, J. M., Guzzi, J., Longchamp, V., Magnenat, S., Mathews, N., Montes de Oca, M., O'Grady, R., Pinciroli, C., Pini, G., Retornaz, P., Roberts, J., Sperati, V., Stirling, T., Stranieri, A., Stutzle, T., Trianni, V., Tuci, E., Turgut, A. E., and Vaussard, F. (2013). Swarmanoid: A novel concept for the study of heterogeneous robotic swarms. *IEEE Robotics Automation Magazine*, 20(4):60–71.

Dorigo, M., Tuci, E., Groß, R., Trianni, V., Labella, T. H., Nouyan, S., Ampatzis, C., Deneubourg, J.-L., Baldassarre, G., Nolfi, S., Mondada, F., Floreano, D., and Gambardella, L. M. (2004). The swarm-bots project. In *International Workshop on Swarm Robotics*, pages 31–44. Springer.

Doursat, R. (2009). Organically grown architectures: Creating decentralized, autonomous systems by embryomorphic engineering. In *Organic Computing*, pages 167–199. Springer.

Doursat, R., Sayama, H., and Michel, O. (2012). *Morphogenetic engineering: Toward programmable complex systems*. Springer.

Doursat, R., Sayama, H., and Michel, O. (2013). A review of morphogenetic engineering. *Natural Computing*, 12(4):517–535.

Driever, W. and Nüsslein-Volhard, C. (1988a). The bicoid protein determines position in the drosophila embryo in a concentration-dependent manner. *Cell*, 54(1):95–104.

Driever, W. and Nüsslein-Volhard, C. (1988b). A gradient of bicoid protein in drosophila embryos. *Cell*, 54(1):83–93.

Driewer, F., Baier, H., and Schilling, K. (2005). Robot–human rescue teams: A user requirements analysis. *Advanced Robotics*, 19(8):819–838.

Durham, A. C. and Ridgway, E. B. (1976). Control of chemotaxis in physarum polycephalum. *The Journal of Cell Biology*, 69(1):218–223.

Dussutour, A., Latty, T., Beekman, M., and Simpson, S. J. (2010). Amoeboid organism solves complex nutritional challenges. *Proceedings of the National Academy of Sciences*, 107(10):4607–4611.

Economou, A. D., Ohazama, A., Porntaveetus, T., Sharpe, P. T., Kondo, S., Basson, M. A., Gritli-Linde, A., Cobourne, M. T., and Green, J. B. A. (2012). Periodic stripe formation by a turing mechanism operating at growth zones in the mammalian palate. *Nature Genetics*, 44(3):348.

Eurobarometer 460, S. (2017). *Attitudes towards the impact of digitisation and automation on daily life*. European Commission.

Ferreira, G. B., Scheutz, M., and Levin, M. (2018). Modeling cell migration in a simulated bioelectrical signaling network for anatomical regeneration. In *Artificial Life Conference Proceedings*, pages 194–201. MIT Press.

Fleming, T. P. and Johnson, M. H. (1988). From egg to epithelium. *Annual Review of Cell Biology*, 4(1):459–485.

Fotaris, P. and Mastoras, T. (2019). Escape rooms for learning: A systematic review. In *ECGBL 2019 13th European Conference on Game-Based Learning*, page 235. Academic Conferences and publishing limited.

Francesca, G., Brambilla, M., Brutschy, A., Trianni, V., and Birattari, M. (2014). Automode: A novel approach to the automatic design of control software for robot swarms. *Swarm Intelligence*, 8(2):89–112.

Friston, K. (2010). The free-energy principle: A unified brain theory? *Nature Reviews Neuroscience*, 11(2):127–138.

Gauci, M., Chen, J., Li, W., Dodd, T. J., and Gross, R. (2014a). Clustering objects with robots that do not compute. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-Agent Systems*, AAMAS '14, page 421–428, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems.

Gauci, M., Chen, J., Li, W., Dodd, T. J., and Groß, R. (2014b). Self-organized aggregation without computation. *The International Journal of Robotics Research*, 33(8):1145–1161.

Gauci, M., Nagpal, R., and Rubenstein, M. (2018). *Programmable Self-disassembly for Shape Formation in Large-Scale Robot Collectives*, pages 573–586. Springer.

Giang, C., Chevalier, M., Negrini, L., Peleg, R., Bonnet, E., Piatti, A., and Mondada, F. (2018). Exploring escape games as a teaching tool in educational robotics. In *International Conference EduRobotics 2016*, pages 95–106. Springer.

Gierer, A. and Meinhardt, H. (1972). A theory of biological pattern formation. *Kybernetik*, 12(1):30–39.

Gilbert, S. F. (2000). *Developmental Biology (sixth edition)*. Sunderland, Massachusetts.

Gray, P. and Scott, S. (1984). Autocatalytic reactions in the isothermal, continuous stirred tank reactor: Oscillations and instabilities in the system A+2B->3B; B->C. *Chemical Engineering Science*, 39(6):1087–1097.

Green, J. B. and Smith, J. (1991). Growth factors as morphogens: Do gradients and thresholds establish body plan? *Trends in Genetics*, 7(8):245–250.

Green, J. B. A. and Sharpe, J. (2015). Positional information and reaction-diffusion: Two big ideas in developmental biology combine. *Development*, 142(7):1203–1211.

Groß, R., Bonani, M., Mondada, F., and Dorigo, M. (2006a). Autonomous self-assembly in a swarm-bot. In Murase, K., Sekiyama, K., Naniwa, T., Kubota, N., and Sitte, J., editors, *Proceedings of the 3rd International Symposium on Autonomous Minirobots for Research and Edutainment (AMiRE 2005)*, pages 314–322. Springer.

Groß, R., Tuci, E., Dorigo, M., Bonani, M., and Mondada, F. (2006b). Object transport by modular robots that self-assemble. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pages 2558–2564. IEEE.

Guo, H., Meng, Y., and Jin, Y. (2011). Swarm robot pattern formation using a morphogenetic multi-cellular based self-organizing algorithm. In *2011 IEEE International Conference on Robotics and Automation*, pages 3205–3210. Springer.

Gurdon, J. and Bourillot, P.-Y. (2001). Morphogen gradient interpretation. *Nature*, 413(6858):797–803.

Gustafson, E. J. (1998). Quantifying landscape spatial pattern: What is the state of the art? *Ecosystems*, 1(2):143–156.

Hamann, H. (2018). Introduction to swarm robotics. In *Swarm Robotics: A Formal Approach*, pages 1–32. Springer.

Hamburger, V. and Hamilton, H. L. (1951). A series of normal stages in the development of the chick embryo. *Journal of Morphology*, 88(1):49–92.

Harbers, M., de Greeff, J., Kruijff-Korbayová, I., Neerincx, M. A., and Hindriks, K. V. (2017). Exploring the ethical landscape of robot-assisted search and rescue. In *A World with Robots*, pages 93–107. Springer.

Hauert, S. and Bhatia, S. N. (2014). Mechanisms of cooperation in cancer nanomedicine: Towards systems nanotechnology. *Trends in Biotechnology*, 32(9):448–455.

Hauert, S., Zufferey, J.-C., and Floreano, D. (2009). Evolved swarming without positioning information: An application in aerial communication relay. *Autonomous Robots*, 26(1):21–32.

Hilder, J., Naylor, R., Rizihs, A., Franks, D., and Timmis, J. (2014). The pi swarm: A low-cost platform for swarm robotics research and education. In *Conference Towards Autonomous Robotic Systems*, pages 151–162. Springer.

Ho, A. M. (2018). Unlocking ideas: Using escape room puzzles in a cryptography classroom. *Primus*, 28(9):835–847.

Horder, T. (2010). *History of Developmental Biology*. American Cancer Society.

Ikemoto, Y., Hasegawa, Y., Fukuda, T., and Matsuda, K. (2005). Gradual spatial pattern formation of homogeneous robot group. *Information Sciences*, 171(4):431–445.

Jaeger, J. and Sharpe, J. (2014). On the concept of mechanism in development. In Minelli, A. and Pradeu, T., editors, *Towards a Theory of Development*, chapter 4, pages 56–78. Oxford University Press Oxford, UK.

Jakobi, N., Husbands, P., and Harvey, I. (1995). Noise and the reality gap: The use of simulation in evolutionary robotics. In *European Conference on Artificial Life*, pages 704–720. Springer.

Jansson, F., Hartley, M., Hinsch, M., Slavkov, I., Carranza, N., Olsson, T. S. G., Dries, R. M., Grönqvist, J. H., Marée, A. F. M., Sharpe, J., Kaandorp, J. A., and Grieneisen, V. A. (2015). Kilombo: A kilobot simulator to enable effective research in swarm robotics.

Jin, Y., Guo, H., and Meng, Y. (2012). A hierarchical gene regulatory network for adaptive multirobot pattern formation. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(3):805–816.

Jirásek, J. E. (2012). *Atlas of human prenatal morphogenesis*. Springer Science & Business Media.

Jones, J. (2011). Influences on the formation and evolution of physarum polycephalum inspired emergent transport networks. *Natural Computing*, 10(4):1345–1369.

Karim, M. E., Lemaignan, S., and Mondada, F. (2015). A review: Can robots reshape k-12 stem education? In *2015 IEEE International Workshop on Advanced Robotics and its Social Impacts (ARSO)*, pages 1–8. IEEE.

Karlebach, G. and Shamir, R. (2008). Modelling and analysis of gene regulatory networks. *Nature Reviews Molecular Cell Biology*, 9(10):770–780.

Keller, R. (2005). Cell migration during gastrulation. *Current Opinion in Cell Biology*, 17(5):533–541.

Kernbach, S., Meister, E., Schlachter, F., Jebens, K., Szymanski, M., Liedke, J., Laneri, D., Winkler, L., Schmickl, T., Thenius, R., Corradi, P., and Ricotti, L. (2008). Symbiotic robot organisms: Replicator and symbrion projects. In *Proceedings of the 8th Workshop on Performance Metrics for Intelligent Systems*, PerMIS '08, page 62–69, New York, NY, USA. Association for Computing Machinery.

Kimmel, C. B., Ballard, W. W., Kimmel, S. R., Ullmann, B., and Schilling, T. F. (1995). Stages of embryonic development of the zebrafish. *Developmental Dynamics*, 203(3):253–310.

Kolling, A., Walker, P., Chakraborty, N., Sycara, K., and Lewis, M. (2016). Human interaction with robot swarms: A survey. *IEEE Transactions on Human-Machine Systems*, 46(1):9–26.

Kondo, S. and Miura, T. (2010). Reaction-diffusion model as a framework for understanding biological pattern formation. *Science*, 329(5999):1616–1620.

Kruijff, G. J. M., Janíček, M., Keshavdas, S., Larochelle, B., Zender, H., Smets, N. J. J. M., Mioch, T., Neerincx, M. A., Diggelen, J. V., Colas, F., Liu, M., Pomerleau, F., Siegwart, R., Hlaváč, V., Svoboda, T., Petříček, T., Reinstein, M., Zimmermann, K., Pirri, F., Gianni, M., Papadakis, P., Sinha, A., Balmer, P., Tomatis, N., Worst, R., Linder, T., Surmann, H., Tretyakov, V., Corrao, S., Pratzler-Wanczura, S., and Sulk, M. (2014). Experience in system design for human-robot teaming in urban search and rescue. In Yoshida, K. and Tadokoro, S., editors, *Field and Service Robotics: Results of the 8th International Conference*, pages 111–125, Berlin, Heidelberg. Springer Berlin Heidelberg.

Kubota, Y. and Ito, K. (2000). Chemotactic migration of mesencephalic neural crest cells in the mouse. *Developmental Dynamics: An Official Publication of the American Association of Anatomists*, 217(2):170–179.

Kuhl, F. P. and Giardina, C. R. (1982). Elliptic fourier features of a closed contour. *Computer Graphics and Image Processing*, 18(3):236–258.

Lee, H. R., Šabanović, S., Chang, W.-L., Nagata, S., Piatt, J., Bennett, C., and Hakken, D. (2017). Steps toward participatory design of social robots: Mutual learning with older adults with depression. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, pages 244–253.

Leite, I., Martinho, C., and Paiva, A. (2013). Social robots for long-term interaction: A survey. *International Journal of Social Robotics*, 5(2):291–308.

Levin, M. (2005). Left–right asymmetry in embryonic development: A comprehensive review. *Mechanisms of Development*, 122(1):3–25.

Levin, M. (2012). Molecular bioelectricity in developmental biology: New tools and recent discoveries: Control of cell behavior and pattern formation by transmembrane potential gradients. *Bioessays*, 34(3):205–217.

Li, H., Wang, T., Wei, H., and Meng, C. (2016). Response strategy to environmental cues for modular robots with self-assembly from swarm to articulated robots. *Journal of Intelligent & Robotic Systems*, 81(3-4):359–376.

Li, M. and Vitányi, P. (2008). *An introduction to Kolmogorov complexity and its applications*, volume 3. Springer.

Litingtung, Y. and Chiang, C. (2000). Specification of ventral neuron types is mediated by an antagonistic interaction between shh and gli3. *Nature Neuroscience*, 3(10):979–985.

Liu, W. and Winfield, A. F. T. (2012). *Distributed Autonomous Morphogenesis in a Self-Assembling Robotic System*, pages 89–113. Springer Berlin Heidelberg, Berlin, Heidelberg.

Liu, W. and Winfield, A. F. T. (2014). Self-assembly in heterogeneous modular robots. In Ani Hsieh, M. and Chirikjian, G., editors, *Distributed Autonomous Robotic Systems*, pages 219–232, Berlin, Heidelberg. Springer Berlin Heidelberg.

Liu, Y., Gao, C., Zhang, Z., Wu, Y., Liang, M., Tao, L., and Lu, Y. (2017). A new multi-agent system to simulate the foraging behaviors of physarum. *Natural Computing*, 16(1):15–29.

López-Pernas, S., Gordillo, A., Barra, E., and Quemada, J. (2019). Examining the use of an educational escape room for teaching programming in a higher education setting. *IEEE Access*, 7:31723–31737.

Magurran, A. E. (1990). The adaptive significance of schooling as an anti-predator defence in fish. In *Annales Zoologici Fennici*, pages 51–66. JSTOR.

Mamei, M., Vasirani, M., and Zambonelli, F. (2004). Experiments of morphogenesis in swarms of simple mobile robots. *Applied Artificial Intelligence*, 18(9-10):903–919.

Marcon, L. and Sharpe, J. (2012). Turing patterns in development: What about the horse part? *Current Opinion in Genetics and Development*, 22(6):578–584.

Marcus, L. F. (1990). Traditional morphometrics. In *Proceedings of the Michigan Morphometrics Workshop*, volume 2, pages 77–122. The University of Michigan Museum of Zoology.

Marlow, F. L. (2010). Maternal control of development in vertebrates. In *Colloquium Series on Developmental Biology*, volume 1, pages 1–196. Morgan & Claypool Life Sciences.

Mathews, N., Christensen, A. L., O'Grady, R., Mondada, F., and Dorigo, M. (2017). Mergeable nervous systems for robots. *Nature Communications*, 8(1):439.

Meinhardt, H. (1982). *Models of biological pattern formation*. Citeseer.

Meinhardt, H. (2009). *The algorithmic beauty of sea shells*. Springer Science & Business Media.

Meinhardt, H. (2012). Turing's theory of morphogenesis of 1952 and the subsequent discovery of the crucial role of local self-enhancement and long-range inhibition. *Interface Focus*, 2(4):407–416.

Meng, Y., Guo, H., and Jin, Y. (2013). A morphogenetic approach to flexible and robust shape formation for swarm robotic systems. *Robotics and Autonomous Systems*, 61(1):25–38.

Miller, J. D. (1983). Scientific literacy: A conceptual and empirical review. *Daedalus*, 112(2):29–48.

Minelli, A. and Pradeu, T. (2014). *Towards a theory of development*. OUP Oxford.

Miyazawa, S., Okamoto, M., and Kondo, S. (2010). Blending of animal colour patterns by hybridization. *Nature Communications*, 1(1):1–6.

Mondada, F., Bonani, M., Raemy, X., Pugh, J., Cianci, C., Klaptocz, A., Magnenat, S., Zufferey, J.-C., Floreano, D., and Martinoli, A. (2009). The e-puck, a robot designed for education in engineering. In *Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions*, volume 1, pages 59–65. IPCB: Instituto Politécnico de Castelo Branco.

Mondada, F., Bonani, M., Riedo, F., Briod, M., Pereyre, L., Rétornaz, P., and Magnenat, S. (2017). Bringing robotics to formal education: The thymio open-source hardware robot. *IEEE Robotics & Automation Magazine*, 24(1):77–85.

Mondada, F., Pettinaro, G. C., Guignard, A., Kwee, I. W., Floreano, D., Deneubourg, J.-L., Nolfi, S., Gambardella, L. M., and Dorigo, M. (2004). Swarm-bot: A new distributed robotic concept. *Autonomous Robots*, 17(2-3):193–221.

Monier, B., Pélissier-Monier, A., and Sanson, B. (2011). Establishment and maintenance of compartmental boundaries: Role of contractile actomyosin barriers. *Cellular and Molecular Life Sciences*, 68(11):1897–1910.

Morelli, L. G., Uriu, K., Ares, S., and Oates, A. C. (2012). Computational approaches to developmental patterning. *Science*, 336(6078):187–191.

Moser, D., Zechmeister, H. G., Plutzar, C., Sauberer, N., Wrbka, T., and Grabherr, G. (2002). Landscape patch shape complexity as an effective measure for plant species richness in rural landscapes. *Landscape Ecology*, 17(7):657–669.

Müller, H.-A. J. (2001). Of mice, frogs and flies: Generation of membrane asymmetries in early development. *Development, Growth & Differentiation*, 43(4):327–342.

Müller, P., Rogers, K. W., Jordan, B. M., Lee, J. S., Robson, D., Ramanathan, S., and Schier, A. F. (2012). Differential diffusivity of nodal and lefty underlies a reaction-diffusion patterning system. *Science*, 336(6082):721–724.

Murphy, R. R. (2004). Human-robot interaction in rescue robotics. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 34(2):138–153.

Murphy, R. R., Tadokoro, S., Nardi, D., Jacoff, A., Fiorini, P., Choset, H., and Erkmen, A. M. (2008). Search and rescue robotics. In *Springer Handbook of Robotics*, pages 1151–1173. Springer.

Murray, J. (1982). Parameter space for turing instability in reaction diffusion mechanisms: A comparison of models. *Journal of Theoretical Biology*, 98(1):143–163.

Murray, J. D. (2003a). *Mathematical Biology I: An Introduction (third edition)*. Interdisciplinary Applied Mathematics. Springer.

Murray, J. D. (2003b). *Mathematical Biology II: Spatial Models and Biomedical Applications (third edition)*. Interdisciplinary Applied Mathematics. Springer.

Nagi, J., Giusti, A., Gambardella, L. M., and Di Caro, G. A. (2014). Human-swarm interaction using spatial gestures. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3834–3841. IEEE.

Nagi, J., Ngo, H., Giusti, A., Gambardella, L. M., Schmidhuber, J., and Di Caro, G. A. (2012). Incremental learning using partial feedback for gesture-based human-swarm interaction. In *2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication*, pages 898–905.

Nagpal, R. and Mamei, M. (2004). Engineering amorphous computing systems. In *Methodologies and Software Engineering for Agent Systems*, pages 303–320. Springer.

Nakagaki, T., Yamada, H., and Tóth, Á. (2000). Intelligence: Maze-solving by an amoeboid organism. *Nature*, 407(6803):470.

Nembrini, J. and Winfield, A. F. T. (2012). *Emergent Swarm Morphology Control of Wireless Networked Mobile Robots*, pages 239–271. Springer.

Nicholson, S. (2015). A recipe for meaningful gamification. In *Gamification in Education and Business*, pages 1–20. Springer.

Nicholson, S. (2016). The state of escape: Escape room design and facilities. *Meaningful Play Conference*.

Nicholson, S. (2018). Creating engaging escape rooms for the classroom. *Childhood Education*, 94(1):44–49.

Niessing, D., Blanke, S., and Jackle, H. (2002). Bicoid associates with the 5-cap-bound complex of caudal mrna and represses translation. *Genes & Development*, 16(19):2576–2582.

Noirot, C. and Darlington, J. P. E. C. (2000). *Termite Nests: Architecture, Regulation and Defence*, pages 121–139. Springer.

Nuñez, I. N., Matute, T. F., Del Valle, I. D., Kan, A., Choksi, A., Endy, D., Haseloff, J., Rudge, T. J., and Federici, F. (2017). Artificial symmetry-breaking for morphogenetic engineering bacterial colonies. *ACS Synthetic Biology*, 6(2):256–265.

O'Grady, R., Christensen, A. L., and Dorigo, M. (2012). *SWARMORPH: Morphogenesis with Self-Assembling Robots*, pages 27–60. Springer.

Oh, H., Shirazi, A. R., and Jin, Y. (2018). Morphogen diffusion algorithms for tracking and herding using a swarm of kilobots. *Soft Computing*, 22(6):1833–1844.

Oh, H., Shirazi, A. R., Sun, C., and Jin, Y. (2017). Bio-inspired self-organising multi-robot pattern formation: A review. *Robotics and Autonomous Systems*, 91:83–100.

Oppenheim, R. W. (1991). Cell death during development of the nervous system. *Annual Review of Neuroscience*, 14(1):453–501.

O'Rahilly, R. (1979). Early human development and the chief sources of information on staged human embryos. *European Journal of Obstetrics & Gynecology and Reproductive Biology*, 9(4):273–280.

Owen, R., Macnaghten, P., and Stilgoe, J. (2012). Responsible research and innovation: From science in society to science for society, with society. *Science and Public Policy*, 39(6):751–760.

Özdemir, A., Gauci, M., and Groß, R. (2017). Shepherding with robots that do not compute. In *Artificial Life Conference Proceedings 14*, pages 332–339. MIT Press.

Özdemir, A., Romanishin, J. W., Groß, R., and Rus, D. (2019). Decentralized gathering of stochastic, oblivious agents on a grid: A case study with 3d m-blocks. In *2019 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)*, pages 245–251. IEEE.

O'Grady, R., Groß, R., Christensen, A. L., and Dorigo, M. (2010). Self-assembly strategies in a group of autonomous mobile robots. *Autonomous Robots*, 28(4):439–455.

Pascalie, J., Potier, M., Kowaliw, T., Giavitto, J.-L., Michel, O., Spicher, A., and Doursat, R. (2016). Developmental design of synthetic bacterial architectures by morphogenetic engineering. *ACS Synthetic Biology*, 5(8):842–861.

Penders, J., Alboul, L., Witkowski, U., Naghsh, A., Saez-Pons, J., Herbrechtsmeier, S., and El-Habbal, M. (2011). A robot swarm assisting a human fire-fighter. *Advanced Robotics*, 25(1-2):93–117.

Pennycuick, C. and Kline, N. (1986). Units of measurement for fractal extent, applied to the coastal distribution of bald eagle nests in the aleutian islands, alaska. *Oecologia*, 68(2):254–258.

Peter, I. S. and Davidson, E. H. (2015). *Genomic control process: Development and evolution*. Academic Press.

Plotze, R. d. O., Falvo, M., Pádua, J. G., Bernacci, L. C., Vieira, M. L. C., Oliveira, G. C. X., and Bruno, O. M. (2005). Leaf shape analysis using the multiscale minkowski fractal dimension, a new morphometric method: A study with passiflora (passifloraceae). *Canadian Journal of Botany*, 83(3):287–301.

Podevijn, G., O'Grady, R., Fantini-Hauwel, C., and Dorigo, M. (2018). Human responses to stimuli produced by robot swarms - the effect of the reality-gap on psychological state. In Groß, R., Kolling, A., Berman, S., Frazzoli, E., Martinoli, A., Matsuno, F., and Gauci, M., editors, *Distributed Autonomous Robotic Systems: The 13th International Symposium*, pages 531–543. Springer International Publishing, Cham.

Podevijn, G., O'Grady, R., Fantini-Hauwel, C., and Dorigo, M. (2016a). Investigating the effect of the reality gap on the human psychophysiological state in the context of human-swarm interaction. *PeerJ Computer Science*, 2:e82.

Podevijn, G., O'grady, R., Mathews, N., Gilles, A., Fantini-Hauwel, C., and Dorigo, M. (2016b). Investigating the effect of increasing robot group sizes on the human psychophysiological state in the context of human–swarm interaction. *Swarm Intelligence*, 10(3):193–210.

Pourmehr, S., Monajjemi, V. M., Vaughan, R., and Mori, G. (2013). "you two! take off!": Creating, modifying and commanding groups of robots using face engagement and indirect speech in voice commands. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 137–142. IEEE.

Rainey, S. and Goujon, P. (2011). Toward a normative ethical governance of technology: Contextual pragmatism and ethical governance. *Journal of Information*, pages 157–179.

Raspopovic, J., Marcon, L., Russo, L., and Sharpe, J. (2014). Digit patterning is controlled by a bmp-sox9-wnt turing network modulated by morphogen gradients. *Science*, 345(6196):566–570.

Reddien, P. W. and Alvarado, A. S. (2004). Fundamentals of planarian regeneration. *Annual Review of Cell and Developmental Biology*, 20:725–757.

Reid, C. R., Latty, T., Dussutour, A., and Beekman, M. (2012). Slime mold uses an externalized spatial "memory" to navigate in complex environments. *Proceedings of the National Academy of Sciences*, 109(43):17490–17494.

Rezeck, P. A., Azpurua, H., and Chaimowicz, L. (2017). Hero: An open platform for robotics research and education. In *2017 Latin American Robotics Symposium (LARS) and 2017 Brazilian Symposium on Robotics (SBR)*, pages 1–6. IEEE.

Riedo, F., Chevalier, M., Magnenat, S., and Mondada, F. (2013). Thymio ii, a robot that grows wiser with children. In *2013 IEEE Workshop on Advanced Robotics and its Social Impacts*, pages 187–193. IEEE.

Rohlf, F. J. and Marcus, L. F. (1993). A revolution morphometrics. *Trends in Ecology & Evolution*, 8(4):129–132.

Romanishin, J. W., Gilpin, K., and Rus, D. (2013). M-blocks: Momentum-driven, magnetic modular robots. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4288–4295. IEEE.

Romanishin, J. W., Mamish, J., and Rus, D. (2019). Decentralized control for 3d m-blocks for path following, line formation, and light gradient aggregation. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4862–4868. IEEE.

Rossiter, J., Winfield, J., and Ieropoulos, I. (2016). Here today, gone tomorrow: Biodegradable soft robots. In Bar-Cohen, Y. and Vidal, F., editors, *Electroactive Polymer Actuators and Devices (EAPAD) 2016*, volume 9798, pages 312 – 321. International Society for Optics and Photonics, SPIE.

Roth, S. (2011). Mathematics and biology: A kantian view on the history of pattern formation theory. *Development, Genes and Evolution*, 221(5-6):255–279.

Rubenstein, M., Ahler, C., Hoff, N., Cabrera, A., and Nagpal, R. (2014a). Kilobot: A low cost robot with scalable operations designed for collective behaviors. *Robotics and Autonomous Systems*, 62(7):966 – 975. Reconfigurable Modular Robotics.

Rubenstein, M., Cornejo, A., and Nagpal, R. (2014b). Programmable self-assembly in a thousand-robot swarm. *Science*, 345(6198):795–799.

Šabanović, S. (2010). Robots in society, society in robots. *International Journal of Social Robotics*, 2(4):439–450.

Sagner, A. and Briscoe, J. (2017). Morphogen interpretation: Concentration, time, competence, and signaling dynamics. *Wiley Interdisciplinary Reviews: Developmental Biology*, 6(4):e271.

Şahin, E. (2005). Swarm robotics: From sources of inspiration to domains of application. In Şahin, E. and Spears, W. M., editors, *Swarm Robotics*, pages 10–20. Springer.

Salazar-Ciudad, I. (2010). Morphological evolution and embryonic developmental diversity in metazoa. *Development*, 137(4):531–539.

Salazar-Ciudad, I. and Jernvall, J. (2004). How different types of pattern formation mechanisms affect the evolution of form and development. *Evolution & Development*, 6(1):6–16.

Salazar-Ciudad, I., Jernvall, J., and Newman, S. A. (2003). Mechanisms of pattern formation in development and evolution. *Development*, 130(10):2027–2037.

Sayama, H. (2009). Swarm chemistry. *Artificial Life*, 15(1):105–114.

Sayama, H. (2010). Robust morphogenesis of robotic swarms [application notes]. *IEEE Computational Intelligence Magazine*, 5(3):43–49.

Sayama, H. (2011). Seeking open-ended evolution in swarm chemistry. In *2011 IEEE Symposium on Artificial Life (ALIFE)*, pages 186–193. IEEE.

Sayama, H. (2018). Seeking open-ended evolution in swarm chemistry ii: Analyzing long-term dynamics via automated object harvesting. In *Artificial Life Conference Proceedings*, pages 59–66. MIT Press.

Schmickl, T. and Crailsheim, K. (2006). Trophallaxis among swarm-robots: A biologically inspired strategy for swarm robotics. In *The First IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, 2006. BioRob 2006.*, pages 377–382. Springer.

Schmickl, T. and Crailsheim, K. (2007). A navigation algorithm for swarm robotics inspired by slime mold aggregation. In Şahin, E., Spears, W. M., and Winfield, A. F. T., editors, *Swarm Robotics*, pages 1–13. Springer.

Schnakenberg, J. (1979). Simple chemical reaction systems with limit cycle behaviour. *Journal of Theoretical Biology*, 81(3):389–400.

Scholes, N. S., Schnoerr, D., Isalan, M., and Stumpf, M. P. (2019). A comprehensive network atlas reveals that turing patterns are common but not robust. *Cell Systems*, 9(3):243–257.

Schranz, M., Umlauft, M., Sende, M., and Elmenreich, W. (2020). Swarm robotic behaviors and current applications. *Frontiers in Robotics and AI*, 7:36.

SciShow (2017). The coming robot swarms. Youtube video accessed at `https://www.youtube.com/watch?v=uAAYivTtOpQ` on 18/11/19.

Senanayake, M., Senthooran, I., Barca, J. C., Chung, H., Kamruzzaman, J., and Murshed, M. (2016). Search and tracking algorithms for swarms of robots: A survey. *Robotics and Autonomous Systems*, 75:422–434.

Shen, W.-M., Will, P., Galstyan, A., and Chuong, C.-M. (2004). Hormone-inspired self-organization and distributed control of robotic swarms. *Autonomous Robots*, 17(1):93–105.

Sheth, R., Marcon, L., Bastida, M. F., Junco, M., Quintana, L., Dahn, R., Kmita, M., Sharpe, J., and Ros, M. A. (2012). Hox genes regulate digit patterning by controlling the wavelength of a turing-type mechanism. *Science*, 338(6113):1476–1480.

Shirk, J. L., Ballard, H. L., Wilderman, C. C., Phillips, T., Wiggins, A., Jordan, R., McCallie, E., Minarchek, M., Lewenstein, B. V., Krasny, M. E., and Bonney, R. (2012). Public participation in scientific research: A framework for deliberate design. *Ecology and Society*, 17(2).

Simis, M. J., Madden, H., Cacciatore, M. A., and Yeo, S. K. (2016). The lure of rationality: Why does the deficit model persist in science communication? *Public Understanding of Science*, 25(4):400–414.

Skilton, M. and Hovsepian, F. (2017). *The 4th industrial revolution: Responding to the impact of artificial intelligence on business*. Springer.

Soloveichik, D. and Winfree, E. (2007). Complexity of self-assembled shapes. *SIAM Journal on Computing*, 36(6):1544–1569.

St Johnston, D. (2015). The renaissance of developmental biology. *PLoS Biology*, 13(5).

St-Onge, D., Levillain, F., Zibetti, E., and Beltrame, G. (2019). Collective expression: How robotic swarms convey information with group motion. *Paladyn, Journal of Behavioral Robotics*, 10(1):418–435.

Stengers, I. (2018). *Another science is possible: A manifesto for slow science*. John Wiley & Sons.

Stilgoe, J., Owen, R., and Macnaghten, P. (2013). Developing a framework for responsible innovation. *Research Policy*, 42(9):1568–1580.

Sturgis, P. and Allum, N. (2004). Science in society: Re-evaluating the deficit model of public attitudes. *Public Understanding of Science*, 13(1):55–74.

Suzanne, M. and Steller, H. (2013). Shaping organisms with apoptosis. *Cell Death & Differentiation*, 20(5):669–675.

Takayama, L., Ju, W., and Nass, C. (2008). Beyond dirty, dangerous and dull: What everyday people think robots should do. In *2008 3rd ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 25–32. IEEE.

Taylor, T., Ottery, P., and Hallam, J. (2007). Pattern formation for multi-robot applications: Robust, self-repairing systems inspired by genetic regulatory networks and cellular self-organisation. *University of Edinburgh., Tech. Rep. EDI-INF-RR-0971*.

Tero, A., Takagi, S., Saigusa, T., Ito, K., Bebber, D. P., Fricker, M. D., Yumiki, K., Kobayashi, R., and Nakagaki, T. (2010). Rules for biologically inspired adaptive network design. *Science*, 327(5964):439–442.

Thenius, R., Dauschan, M., Schmickl, T., and Crailsheim, K. (2011). Regenerative abilities in modular robots using virtual embryogenesis. In Bouchachia, A., editor, *Adaptive and Intelligent Systems*, pages 227–237. Springer.

Thomas, D. and Kernevez, J. P. (1976). Analysis and control of immobilized enzyme systems. In *International Symposium on Analysis and Control of Immobilized Enzyme Systems (1975: Compiegne, France)*. North-Holland.

Tickle, C. (2006). Making digit patterns in the vertebrate limb. *Nature Reviews Molecular Cell Biology*, 7(1):45–53.

Tsarouchi, P., Makris, S., and Chryssolouris, G. (2016). Human–robot interaction review and challenges on task planning and programming. *International Journal of Computer Integrated Manufacturing*, 29(8):916–931.

Turing, A. M. (1952). The chemical basis of morphogenesis. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 237(641):37–72.

Tzoumas, G. (2019). Firefighting swarm robotics. Master's thesis, University of Bristol.

Ueda, T., Muratsugu, M., Kurihara, K., and Kobatake, Y. (1976). Chemotaxis in physarum polycephalum: Effects of chemicals on isometric tension of the plasmodial strand in relation to chemotactic movement. *Experimental Cell Research*, 100(2):337 – 344.

Valzania, G. and WRLD3D (2019). Indoor map. WRLD3D website accessed at `https://www.wrld3d.com/blog/creating-indoor-maps-6-key-steps-for-beginners/` on 11/11/19.

Vassev, E., Sterritt, R., Rouff, C., and Hinchey, M. (2012). Swarm technology at NASA: Building resilient systems. *IT Professional*, 14:36–42.

Vörös, A. I. V. and Sárközi, Z. (2017). Physics escape room as an educational tool. In *AIP Conference Proceedings*, volume 1916, page 050002. AIP Publishing LLC.

Wei, H., Cai, Y., Li, H., Li, D., and Wang, T. (2010). Sambot: A self-assembly modular robot for swarm robot. In *2010 IEEE International Conference on Robotics and Automation*, pages 66–71. IEEE.

Werfel, J., Petersen, K., and Nagpal, R. (2014). Designing collective behavior in a termite-inspired robot construction team. *Science*, 343(6172):754–758.

Wijgerde, M., McMahon, J. A., Rule, M., and McMahon, A. P. (2002). A direct requirement for hedgehog signaling for normal specification of all ventral progenitor domains in the presumptive mammalian spinal cord. *Genes & Development*, 16(22):2849–2864.

Wilson, S., Gameros, R., Sheely, M., Lin, M., Dover, K., Gevorkyan, R., Haberland, M., Bertozzi, A., and Berman, S. (2016). Pheeno, a versatile swarm robotic research and education platform. *IEEE Robotics and Automation Letters*, 1(2):884–891.

Winfield, A. F. T., Franco, M. P., Brüggemann, B., Castro, A., Limon, M. C., Ferri, G., Ferreira, F., Liu, X., Petillot, Y. R., Röning, J., Schneider, F. E., Stengler, E., Sosa, D., and Viguria, A. (2016). Eurathlon 2015: A multi-domain multi-robot grand challenge for search and rescue robots. In *TAROS*. Springer.

Winfield, A. F. T., Harper, C. J., and Nembrini, J. (2004). Towards dependable swarms and a new discipline of swarm engineering. In *International Workshop on Swarm Robotics*, pages 126–142. Springer.

Winfield, A. F. T. and Jirotka, M. (2017). The case for an ethical black box. In Gao, Y., Fallah, S., Jin, Y., and Lekakou, C., editors, *Towards Autonomous Robotic Systems*, pages 262–273, Cham. Springer International Publishing.

Winfield, A. F. T. and Jirotka, M. (2018). Ethical governance is essential to building trust in robotics and artificial intelligence systems. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 376(2133):20180085.

Winkle, K., Caleb-Solly, P., Turton, A., and Bremner, P. (2019). Mutual shaping in the design of socially assistive robots: A case study on social robots for therapy. *International Journal of Social Robotics*, pages 1–20.

Wolpert, L. (1969). Positional information and the spatial pattern of cellular differentiation. *Journal of Theoretical Biology*, 25(1):1–47.

Wolpert, L. (1971). Positional information and pattern formation. *Current Topics in Developmental Biology*, 6(6):183–224.

Wolpert, L., Tickle, C., and Arias, A. M. (2015). *Principles of development*. Oxford university press.

Yanco, H. A., Baker, M., Casey, R., Keyes, B., Thoren, P., Drury, J. L., Few, D., Nielsen, C., and Bruemmer, D. (2006). Analysis of human-robot interaction for urban search and rescue. In *Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics*, pages 22–24. Citeseer.

Yang, G.-Z., Bellingham, J., Dupont, P. E., Fischer, P., Floridi, L., Full, R., Jacobstein, N., Kumar, V., McNutt, M., Merrifield, R., Nelson, B. J., Scassellati, B., Taddeo, M., Taylor, R., Veloso, M., Wang, Z. L., and Wood, R. (2018). The grand challenges of science robotics. *Science Robotics*, 3(14).

Yim, M., Shen, W.-M., Salemi, B., Rus, D., Moll, M., Lipson, H., Klavins, E., and Chirikjian, G. S. (2007). Modular self-reconfigurable robot systems [grand challenges of robotics]. *IEEE Robotics & Automation Magazine*, 14(1):43–52.

Yuan, J. and Yankner, B. A. (2000). Apoptosis in the nervous system. *Nature*, 407(6805):802–809.

Zambonelli, F. and Mamei, M. (2005). Spatial computing: An emerging paradigm for autonomic computing and communication. In Smirnov, M., editor, *Autonomic Communication*, pages 44–57, Berlin, Heidelberg. Springer Berlin Heidelberg.

Zelditch, M. L., Swiderski, D. L., and Sheets, H. D. (2012). *Geometric morphometrics for biologists: A primer*. Academic Press.